Modelling of light propagation in mixed Diffusing and Non-Diffusing Domains

Student:Jason Riley

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

of the

University of London.

Department of Computer Science

University College London

2005

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

Abstract

The aim of this research has been to produce a new model for light transport, which handles diffusing and non-scattering domains simultaneously. The drive for this research comes from a field within medical imaging called Optical Tomography. Optical Tomography uses the behaviour of infra-red light passing through tissue to identify its optical parameters, from which clinical measures such as blood volume and oxygenation can be derived.

A common model for light transport in Optical Tomography is the Diffusion Approximation to the Radiative Transfer Equation, which is appropriate for diffusing media. However in certain tissues such as the Cerebro-Spinal Fluid, this approximation breaks down, due to the non-scattering effect of this fluid. One alternative to this has been to use a full transport model throughout the domain, however this is computationally expensive.

We will introduce a novel 3D model for handling light transport in diffusing domains containing non-scattering spaces. We show a hybridisation of methods from optical and neutron transport theory with further methods taken from computer graphics. We illustrate the validity of the model by comparison with numerical and physical models and use this to show how the clear regions in the head will effect neo-natal imaging.

We will show the model we develop to be as accurate as more expensive full transport models for diffusing domains with non-scattering spaces. We use this to illustrate the effects such non-scattering inclusions have on the problem of Optical Tomography of the neo-natal head.

Contents

List of Figures

List of Tables

Acknowledgements

In completion of this thesis there are those I must thank formally for their direct contributions to my research. Of these I hope I catch you all, and apologise if I miss someone. Then there are those who must be thanked for informal contributions, ranging from proof reading, de-stressing chats, shoulders to cry on, etc.

So first I must thank my supervisors Simon, Yiorgos and Celine for their various and prolonged help and tolerance over the years. Next must come the internal assessors at interim stages, Soren, Jem and Mel for their interest in reading and questioning of my work allowing me to understand and present it better. I must thank the graduate school for their funding of the first three years of my thesis and conference funding, and also Simon and Bernard for putting me forward for this.

Next to all the members of the group past and present for making this possible by their direct/indirect contributions: Simon, Jem, Martin, Hamid, Beth, Jorge, Jan, Adam, Thanasis, Steve, Abdel and the rest. Of you all I must single out Jan due to his working with similar problems and his willingness to help all the people in the lab produce. Also Steve, amongst many, for proof reading and helping me express my work so others can understand it. Martin also gets a mention as without his development and management of TOAST none of us would get half as far as we do.

OK the informal thanks now, guys you've been great, all of the above, Olivier and Maia for helping me find my feet as a PhD student. Then there's the family for financial support since my grant ran out and generally being there. Now the rest er.. well you know who you are. Oh and one final special thanks to Thanasis for taking me to Greece a few times in the last years of this work and rescuing my sanity and drive to complete this.

List of Publications

$\hat{3}$D Optical Tomography in the presence of Void Regions. J. Riley, H. Dehghani, M. Schweiger,S.R. Arridge, J. Ripoll and M. Nieto-Vesperinas, Optics Express(7,13), p462-467, 2000.

The Radiosity-Diffusion Model in 3D. J.Riley et al. Proceedings of the SPIE Vol 4431 - Photon Migration, Optical Coherence Tomography and Microscopy, p153-164, 2001.

Light Transport in Scattering Domains containing Non-Scattering Spaces. J. Riley, E.M.C. Hillman, J.C. Hebden and S.R. Arridge, International Workshop - 'Comptational Problems of Electrical Engineering', p215-218, 2002, Zakopane, Poland.

Optical tomography of a realistic neonatal head phantom. A P Gibson, R Md Yusof, H Dehghani, J Riley, N Everdell, R Richards, J C Hebden, M Schweiger, S R Arridge and D T Delpy (2003), Applied Optics 42(16), p1-8.

Analysis of light propagation in diffusive media with non-scattering regions using 3D BEM. J. Sikora, J. Riley, S. Arridge and J.Ripoll, Technical Report, University College London, 2003.

A method for generating patient-specific finite element meshes for head modelling. A P Gibson, J Riley, M Schweiger, J C Hebden, S R Arridge, D T Delpy (2003), Phys. Med. Biol. 48, p481-495

Linear and non-linear reconstruction for optical tomography of phantoms with non-scattering regions. A P Gibson, J C Hebden, J Riley, N Everdell, M Schweiger, S R Arridge and D T Delpy, (in press), Appl. Opt.

Diffusion regularisation methods of the non-linear inverse problem for diffuse optical tomography. A. Douiri, M. Schweiger, J. Riley and S. R. Arridge, (in press) Proceedings of the 5th International Conference on Inverse Problems in Engineering: Theory and Practice, July 2005.

Nomenclature

Symbol	Meaning of Symbol	Units
Ω	A domain in space $\mathbb{R}^n$	
$\partial\Omega$	The boundary of domain Ω	
$\overline{\Omega}$	A Non-Scattering sub-domain of a diffusing domain Ω	
$\partial\overline{\Omega}$	The boundary of $\overline{\Omega}$	
$\hat{\nu}$	Outward directed normal of $\partial\Omega$ or $\partial\overline{\Omega}$	
$\mathbf{r}$	Point in space ($\mathbf{r} \in \Omega$)	
$\mathbf{m}$	Point in space ($\mathbf{m} \in \partial\Omega$ or $\partial\overline{\Omega}$)	
$\hat{\mathbf{s}}$	Direction of travel ($\hat{\mathbf{s}} \in S^{n-1}$)	
t	Time	ps
c	Speed of Light	mm ps^{-1}
μ_{a}	Absorption Coefficient	mm^{-1}
μ_{s}	Scattering Coefficient	mm^{-1}
μ'_{s}	Reduced Scattering Coefficient	mm^{-1}
μ_{t}	Transport Coefficient	mm^{-1}
$\Theta(\mathbf{r}, \hat{\mathbf{s}}, \hat{\mathbf{s}}')$	Scattering probability distribution (Phase Function)	

Symbol	**Meaning of Symbol**	**Units**
$\phi(\mathbf{r}, \hat{\mathbf{s}}, t)$	radiance	photons mm^{-3} sr^{-1} ps^{-1}
$\Phi(\mathbf{r}, t)$	Photon Density	photons mm^{-3} ps^{-1}
$q(\mathbf{r}, \hat{\mathbf{s}}, t)$	Photon Source	photons mm^{-3} sr^{-1} ps^{-1}
$\rho_{brdf}\mathbf{m}, \hat{\mathbf{s}}, \hat{\mathbf{s}}$	Bi-Directional Reflection Distribution Function (BRDF)	
(η, χ)	Equatorial and azimuthal angles on a sphere	radians

Chapter 1

Introduction

This research has aimed at investigating ways to model light transport in diffusing domains containing non-diffusing spaces or sub-domains. The primary drive for this research has been from the field of Optical Tomography, a tool being developed to monitor blood volume and oxygenation. In this thesis we will introduce the particular problem we have worked on and show how we have developed a model for light transport in our given domain, by drawing on work and expertise from various disciplines. Our aim has been to produce a model useful in Optical Tomography and our work has mainly focused on an efficient approach to the light transport problem in this context.

In this chapter we will briefly introduce the concepts of Light Transport and its relevance in medical imaging and in particular for Optical Tomography. We will then go on to outline the overall structure of this thesis and conclude by summarising the goals and achievements to be outlined in this thesis.

1.1 The Light Transport Problem

Light transport can be considered in two ways. It is commonly understood we may consider it either as a wave problem or as a particle problem. It is interesting to remember that there are experiments which indicate that actually light behaves as both - some showing results that can only be explained by a wave phenomena (Huygens/Young) others results only explicable by particle behaviour (Newton). In this research we will consider light as a particle phenomenon.

1.1.1 Light Transport Theory

Light transport can be described using the Radiative Transfer Equation (RTE):

$$\begin{aligned} \frac{1}{c}\frac{d\phi(\mathbf{r},\hat{\mathbf{s}},t)}{dt} &+ \hat{\mathbf{s}}.\nabla\phi(\mathbf{r},\hat{\mathbf{s}},t) + \mu_{\mathrm{t}}(\mathbf{r})\phi(\mathbf{r},\hat{\mathbf{s}},t) \\ &= \mu_{\mathrm{s}}(\mathbf{r})\int \Theta(\mathbf{r},\hat{\mathbf{s}},\hat{\mathbf{s}}')\phi(\mathbf{r},\hat{\mathbf{s}}',t)d\hat{\mathbf{s}}' + q(\mathbf{r},\hat{\mathbf{s}},t), \end{aligned} \tag{1.1}$$

a balance equation for particle transport which describes radiance, $\phi(\mathbf{r},\hat{\mathbf{s}},t)$, at a point in a given direction at a specific time in terms of the scattering coefficient, $\mu_{\mathrm{s}}(\mathbf{r})$, the extinction/transport coefficient, $\mu_{\mathrm{t}}(\mathbf{r})$, the phase function, $\Theta(\mathbf{r},\hat{\mathbf{s}},\hat{\mathbf{s}}')$, and the source term, $q(\mathbf{r},\hat{\mathbf{s}},t)$. This equation is used in many fields from atmospherics, astronomical physics, neutron transport in nuclear reactor physics and also light transport[1, 19, 20, 30, 47, 48]. Historically light propagation is treated by various solutions to the full RTE and also to a variety of approximations to it. In part I of this thesis, we will discuss various approaches to solving the full transport equation, outlining a progression towards a model which provides one of the common approximations for diffusive media - the Diffusion Approximation (DA). We will show that this model has some limitations which we wish to avoid but has the advantage of lower computational costs over full models.

We will also discuss how light transport is modelled in Computer Graphics. It is usually done using Kajiya's Rendering Equation (KRE) [51], which represents a stationary radiance field at the surfaces of a non-participating medium (i.e. no scattering or absorption). Surface reflections are modelled by a probability function called the Bi-directional Reflection Distribution Function (BRDF), $\rho_{brdf}(\mathbf{m},\hat{\mathbf{s}},\hat{\mathbf{s}}')$. Recently in graphics other fuller models have been introduced to provide more realistic image synthesis [68, 76]. We will discuss the ideas behind these and contrast them to techniques in more classical light transport. The results presented in graphics do have a common feature, that of seeking a realistic looking image, not necessarily an accurate one. This has led us to an investigation of the need for accuracy in an application such as medical imaging, which provides interesting results.

1.1.2 Light Transport in Medical Optical Imaging

Transport theory has been used previously in medical imaging for techniques such as Positron Emission Tomography (PET), Single Photon Emission Computed Tomogra-

phy (SPECT) and Computed Tomography (CT). These all use a simple Radon Transform to obtain an image, as at the wavelengths of light involved, we have very low or no scatter present in the medium and are interested primarily in the attenuation by absorption. Our interest has stemmed from its use in a relatively new field called Optical Tomography (OT), (see for example [3, 26, 43, 66]). OT is a modality where we hope to obtain information about tissue properties from their response to infra-red light, which is highly scattered in tissue, hence we must go further than the Radon Transform.

The idea is that blood has different optical properties with respect to infra-red light depending on its state of oxygenation, as shown in figure 1.1. In OT it is often common to use wavelengths either side of the cross-over of the oxygenated/deoxygenated curves, for instance MONSTIR [80] the UCL system uses the wavelengths of 780 and 820nms. With information about the scattering and absorbing properties of tissue we can determine its blood volume and oxygenation. Full details of the potential clinical data we can extract are given in [44], as an example here we give the equations for the fractional oxygen saturation ($S = \frac{[Hb0_2]}{[Hb0_2][Hb]}$ where $[Hb0_2]$ is the amount of oxygenated blood and $[Hb]$ is the amount of deoxygenated blood), and the fractional blood volume ($V = \frac{\text{volume of whole blood in tissue}}{\text{total tissue volume}}$):

$$\begin{aligned} \mu_{a,\lambda_1} &= (1-V)\mu_{a,bg,\lambda_1} + V(\mathcal{E}_{Hb0_2,\lambda_1}S + (1-S)\mathcal{E}_{Hb,\lambda_1}) \\ \mu_{a,\lambda_2} &= (1-V)\mu_{a,bg,\lambda_2} + V(\mathcal{E}_{Hb0_2,\lambda_2}S + (1-S)\mathcal{E}_{Hb,\lambda_2}) \end{aligned} \tag{1.2}$$

where μ_{a,bg,λ_1} and μ_{a,bg,λ_2} are the background tissue parameters at our given location and $\mathcal{E}_{Hb0_2,\lambda_1}$, etc are the molar extinction coefficients for fully oxygenated and fully deoxygenated blood at the two wavelengths. We note at this point that the tissue values are usually estimated. We can derive from these relationships equations for S and V to give us clinically useful values. For the rest of this thesis we will consider the obtaining of the optical parameters as these are what we can obtain directly from our observations.

Two main areas of interest are breast imaging and neo-natal brain imaging. In this work we are more concerned with the latter case. Human tissue (e.g. flesh,bone) can normally be treated as a diffusing media and therefore we can model light transport within it using the DA to the RTE:

$$\frac{1}{c}\frac{\partial}{\partial t}\Phi(\mathbf{r},t) - \nabla\cdot\kappa(\mathbf{r})\nabla\Phi(\mathbf{r},t) + \mu_a(\mathbf{r})\Phi(\mathbf{r},t) = q_0(\mathbf{r},t) \;\; \forall \mathbf{r}\in\Omega \tag{1.3}$$

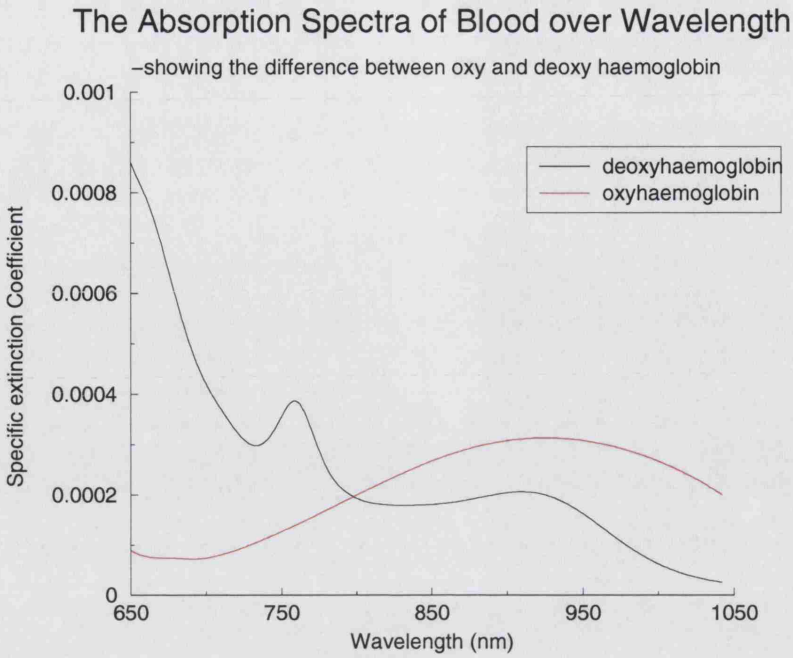


Figure 1.1: A graph illustrating the optical properties of oxygenated/de-oxygenated blood for different wavelengths.

Here we introduce the diffusion parameter, $\kappa = \frac{1}{3(\mu_s' + \mu_a)}$, the absorption parameter, $\mu_a(\mathbf{r})$, and the isotropic source term, $q_0(\mathbf{r}, t)$.

In the case of the head however we have several tissue regions where this equation will fail. One of these is the case of the cerebro-spinal fluid (CSF) regions. These are the sub-arachnoid space (SAS) which surrounds the brain and the ventricles within it. These regions are illustrated in figure 1.2, which shows a slice from a Magnetic Resonance Image (MRI) containing these structures.

The problem in the case of the CSF is that it is a non-scattering region, which is a violation of one of the assumptions of the DA, that $\mu_a \ll \mu_s$. Previous work on this region has used full transport models to model the whole domain in order to cope with this problem [26, 43, 52]. We have looked at extending a 2D model, the Radiosity Diffusion Model (RDM), which uses suitable approximations in each sub-domain (presented in [4]) to 3D to model this problem. A large part of this research has focused on making this problem in 3D tractable and efficient on standard machines. The idea is to minimise the computational expense by using the best approximation to each component of the domain and thereby eliminating the expense of a full model

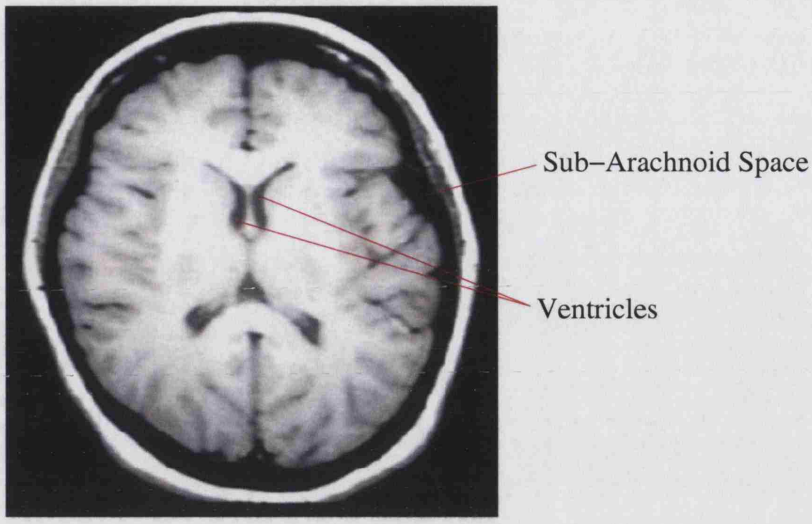


Figure 1.2: A figure illustrating the locations of CSF in the head, using an MRI slice to show the sub-arachnoid space and the ventricles.

everywhere.

One area of investigation we have conducted is the effect of such regions on the data, both in terms of the clear layer and the ventricles. There are two reasons for this, firstly to examine the need for models such as the RDM for OT and secondly for comparison of other models for handling clear layers (such as those given in [13]). We conclude that such models are necessary and that a layer model is not sufficient for modelling. We also show that highly accurate domain boundaries may be necessary for direct imaging - i.e. recovery of brain images from a single data set, although a reasonable approximation may be sufficient when two data sets are available: a background and difference data set. In the optical case there are various ways we can achieve such difference data sets, we may chose to model a classic 'evoked response' i.e. a resting and active brain state, alternatively changes in oxygenation may be achieved by varying the babies O_2CO_2 air mix or finally we may use an object such as an intralipid filled balloon to provide a background or reference data set.

1.1.3 Light Transport in Computer Graphics

In computer graphics we see that light transport is typically rendered in the absence of volume participating media. In other words we deal with light travelling between surfaces surrounding a volume with no scattering or absorbing effects within it (the conditions for Kajiya's Rendering Equation (KRE) to work). Two classic methods

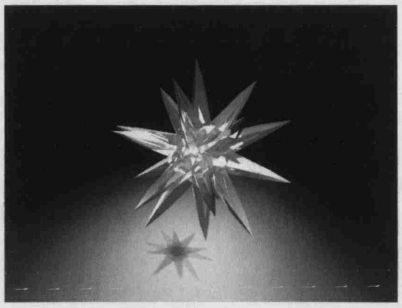


Figure 1.3: A typical Ray Tracing image, taken from the web, see http://www.astro.lu.se/ stefans/povray.html.

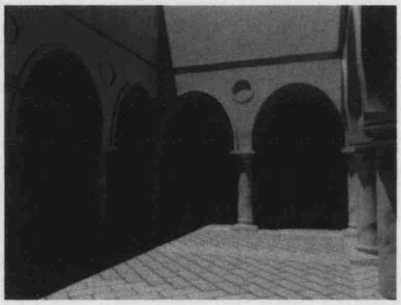


Figure 1.4: A typical Radiosity image taken from the web, see http://lightwave.sns.hu/galeria/gabe/radiosity.jpg.

exist for modelling KRE, ray-tracing and radiosity, which represent models to render the two extremes of the bi-directional reflection distribution function (BRDF). Ray-Tracing models the case of perfect specular reflection at surfaces, or mirrors, and is illustrated in figure 1.3. Radiosity models the case of Lambertian or diffuse reflection at the surfaces, such as matt painted walls, and is illustrated in figure 1.4.

This begins the study of light representations in graphics and details of these techniques can be found in most graphics texts (e.g. [94]). However work has continued from here to include volume effects, either beneath the surface, for the purpose of rendering caustics (e.g. marble) or for true volume effects in the scene, such as smoke or fog. A variety of approaches have been considered from direct methods, like the zonal approach [78, 91], to stochastic models such as Monte-Carlo (e.g.[95]) and Photon-Mapping (e.g. [49]). Reviews of these approaches can be found in [68, 77], where the merits of the various approaches are discussed, we will discuss some of the more relevant of these models in chapter 4.

1.2 Light Transport in mixed diffusing and non-diffusing domains

As we have outlined above we seek to develop a tool for diffusive domains containing non-diffusive regions. The model can also be applied to non-diffusive domains containing diffusive regions. The idea has been to develop a model which can act as the basis for devising a hybrid which uses suitable approximations to transport in all of its sub-regions. This would obviously be applicable in any of the transport disciplines already mentioned, though its usefulness would depend on the individual transport problems being handled. In the first instance we have developed this model with a primary interest in medical imaging.

Our aim has been twofold in this work, first our objective was to produce a direct and numerically accurate model for solving this transport problem. Once this was achieved (within numerical constraints of the modelling process), we aimed to examine the effects of varying the model on both efficiency and accuracy to examine the usefulness and stability of the model.

1.2.1 A tool for Optical Imaging

The Radiosity Diffusion Model (RDM) theoretically represents one of the most efficient approaches to solving the light transport problem for a mixed diffusing and non-scattering domain. Having developed a model which solves this problem there are many things we can address. First we must show that such a model is valid, then we can use it to understand light transport in such domains.

In order to validate the model we have made two comparisons. First we show that the RDM is numerically equivalent to solving a full transport model in the same domain (in this case a Monte-Carlo simulator). The next stage was a physical validation, comparing the output of the RDM to data measured on a known object using the MONSTIR system at UCL [80]. This has allowed us to show that our model is not just numerically correct, but also physically valid. This allows us to use it as a tool to examine optical behaviour.

In terms of optical imaging we have carried out a selection of assessments to judge the need for models such as the RDM and the effects of such models on the potential for imaging. We will illustrate the comparison of the RDM with other models, showing

the strengths and weaknesses of each. We will show that in some situations it may be possible to use simpler models to handle clear regions, but that as always simplifying the model has its limitations and such models may not always describe the light transport in sufficient detail for our purposes.

1.3 Overview of the Thesis

This thesis will be presented in three parts, the background, the development of the RDM, and the results and conclusions. We will outline the content of the parts below and finally give a summary of what we have achieved in this research and the implications of these outcomes.

1.3.1 Structure of the Thesis

The first part of this thesis is dedicated to a more detailed review of background material in this discipline. This part will cover work from the underlying mathematics required to develop the models we use in this research, moving through an introduction to the RTE and then a background of theory from both classical transport fields and computer graphics. In this section we will try to show how ideas for modelling the RTE have developed following a fairly natural progression, and how this model becomes an extension of this development.

In the second part of this thesis we will present the body of the work done to develop the RDM in 3D. We will illustrate the simple model used in 2D and show how this has been evolved to form a practical working 3D model. Here we will show some simple analytic models we have developed for testing the model, and progress to show some of the various approaches we have experimented with to improve the models efficiency; both in terms of improving the computational aspect, but also by trying to reduce the numerical overheads by reducing our unknowns. In this work these two factors have played a key role in providing a model which is sufficient to the problem, and can be solved in reasonable computational memory and time. We illustrate the approaches we have taken to reducing these factors to produce tractable problems.

In the final part of this thesis we will present the results of our work. We will start by validation studies of the basic model, proving the RDM to be correct. Having done this we will go on to compare the various approaches developed in studying the model,

trying to indicate the robustness of the approach. Finally we will show an investigation of the implications of the model for work in fields such as OT giving a description of its uses and implications.

1.3.2 Goals and Achievements of the Work

In this work we will describe a novel model for handling light transport in diffusive domains containing non-scattering regions in 3D. We will show validation of this model numerically against other models and physically against various phantoms. We will show that this model is stable to geometric change as expected, but that an accurate field model is dependent on a good geometric knowledge of our problem domain. Further we will demonstrate that such models are necessary to provide accurate solutions to the domain, but that simpler models may be able to approximate the broad effects of non-scattering inclusions in diffusive media.

We will demonstrate various approaches to solving the components of the radiosity component of the RDM, including a novel, though limited domain geometric visibility algorithm. Previous work [99] has shown anecdotally that different approximations to the Form Factor (an expression which expresses the BRDF for the Radiosity domain) produce different qualities of approximation. We provide further results which support this, but also show that we must consider the entire system to be sure of choosing the appropriate approximation for our model.

We will illustrate various approaches to modelling geometric complexity, including mixed order meshing and a novel simplified meshless model. We will find that whilst mixed order elements are helpful in modelling they are not useful in reducing our computational burden. We will show that our simplified meshless approach represents a reasonable approximation for handling our domains. At the moment it carries the caveat that an assumption is made regarding the surface representation, we will discuss in the section on future work improvements which would remove the need for this assumption and move it towards being a full meshless model.

In terms of Optical Tomography we will demonstrate that diffusion models and assumptions are insufficient to the problem at hand. We will illustrate that non-scattering models such as the RDM are necessary to handle these domains. We will compare our model with other simpler models showing that some, such as the Modified Diffusion

Model (MDM) [13], are sufficient given certain geometries, we conjecture they could be extended to more general geometries, but this is future work. We will further show that if we are to reconstruct clear regions, some features which may be important to the problem are only present in more sophisticated models such as the RDM. We will go on to show that whilst reconstructing from a single data set will require highly accurate knowledge of our geometry we should be able to obtain meaningful results from less accurate geometries in the presence of two data sets: a background and a difference set.

Part I

Background

Chapter 2

Mathematics and Meshing - Tools used in this Thesis

In this chapter we will introduce the basic mathematical tools required in this thesis. These will be how we represent functions in space, how we solve partial differential equations and an introduction to spherical harmonics and the Legendre polynomials. We will also be examining how we model our domains, that is how we subdivide our domains in 2D and 3D.

2.1 Mathematics

In this section we outline the mathematics which we will use to model the RTE. We provide some elementary definitions and symbols which we will use throughout this text.

2.1.1 Approximating functions

If we are interested in solving the problem of the photon distribution in space, we must be able to form a representation of a function in space and angle.

Functions to be represented

We will be considering functions in a domain Ω in $\mathbb{R}^n$ (where n gives us the dimension of the domain) with boundary $\partial\Omega$. We would like to be able to represent a function of order N with continuity C^{N-1}.

Definition 2.1 *A function is said to be C^N continuous if the Nth derivative of the function is continuous.*

Subdividing the Domain

The first thing to consider when representing a function is how to subdivide the domain. The classical approaches will divide the domain into some set of T subregions τ_t, such that $\tau_t \in \Omega$, $\cup_{t=1}^{T} \tau_t \equiv \Omega$ where $\{t = 1, 2, ...T\}$. This can be done in either a structured (regular) format or an unstructured (irregular) format. The choice of structured or unstructured format will depend on the type of domain and also the method used to represent the function. Some details of methods for subdividing the domain are given in section 2.2 on mesh generation.

2.1.2 Finite Element Method

The Finite Element Method (FEM) is typically (though not necessarily) an unstructured approach to approximating functions. It represents the function as a set of samples which are interpolated by basis functions:

$$f(\mathbf{r}) \approx \tilde{f}(\mathbf{r}) = \sum_{i=1}^{N} f_i u_i(\mathbf{r}) \tag{2.1}$$

where N is the number of samples, f_i is the value of f at the sample point, or node (the i-th node being represented $\mathbf{x}_i$), and $u_i(\mathbf{r})$ is the basis function associated with sample i.

Basis Functions

There are many types of basis function which can be used for Finite Elements. Basis functions typically have certain properties, these are:

- A given basis function will have support only over the elements containing its node. ($u_i(\mathbf{r}) > 0$ iff $\mathbf{r} \in \tau$ and $\mathbf{x}_i \in \tau$)
- A given basis function is 1 at its own node and 0 at all other nodes. ($u_i(\mathbf{r}) = 1$ iff $\mathbf{r} \equiv \mathbf{x}_i$, 0 if $\mathbf{r} \equiv \mathbf{x}_k$ and $k \neq i$)
- All basis functions sum to 1 at all points. ($\sum_{i=1}^{N} u_i(\mathbf{r}) = 1 \; \forall \, \mathbf{r} \in \Omega$)
- Basis functions ensure continuity of the function inside the element.

The maths library of the software used for our model (Time resolved Optical Absorption and Scattering Tomography(TOAST) [85]) uses linear, quadratic and cubic basis functions on tetrahedra and trilinear basis function in voxel elements. The tetrahedral

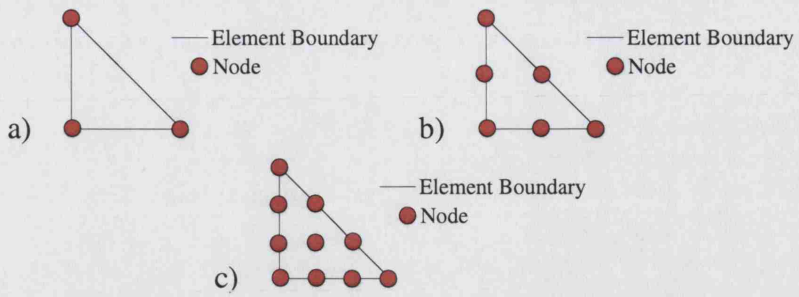


Figure 2.1: A diagram illustrating 2D elements showing triangles with linear (a), quadratic (b), and cubic (c) node placements.

element basis functions have the advantage that the calculation of integrals of the basis functions and their derivatives are available in closed form.

Sample Points

The location of sample points is related to the order of the FE approximation and the elements used. In our case we typically use triangular elements in 2D and tetrahedral elements in 3D. For linear elements we use the vertices as the nodes, and increased sampling is achieved by further subdividing the element in a regular manner, this is illustrated for linear, quadratic and cubic triangles in figure 2.1.

Iso-Parametric Elements

With higher order elements (greater than linear) comes an extra possibility, that is, to describe the shape of the element boundary as being curved. This is done by using the same basis functions to describe the shape of the element as are used to describe the function across it. This leads to elements which have a local C^{N-1} continuous shape, where N is the element order. Such iso-parametric elements are limited in two ways:

- They only have local higher order continuity in shape
- As they use specific basis functions the set of shapes which can be described is limited.

There are of course problems associated with iso-parametric elements, primarily they require more complicated calculation of the integrals of their basis functions. As an extra mapping is involved we cannot use the closed form used for standard elements and must revert to, more computationally costly, quadrature rules for integration. An illustration of the shape of an iso-parametric element is given in figure 2.2.

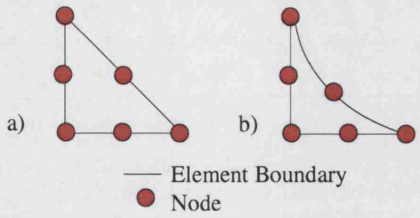


Figure 2.2: A diagram illustrating the capacity of iso-parametric elements to represent curved boundaries (a) showing the classic element shape and (b) one potential iso-parametric element of equivalent order.

2.1.3 Partial Differential Equations

In this work we aim to model the Diffusion Approximation to the RTE, this is a Partial Differential Equation (PDE) - an equation of a function and its derivatives. In order to work with this we must have some understanding of how such PDE's are handled. In this section we briefly outline some of the important techniques which will be used in this work.

Boundary Conditions

When solving a PDE the first thing we must consider is what occurs at the boundary of the domain. There are various classical types of Boundary Condition (BC) each with a distinct physical meaning, the choice of BC will depend largely on the problem. A few typical choices are listed below (Note: here the symbol $\mathbf{m}$, where $\mathbf{m} \equiv \mathbf{r} \in \partial\Omega$ has been introduced):

- Dirichlet - The Dirichlet Boundary conditions are where we know the function on the boundary, i.e. $f(\mathbf{m}) = y(\mathbf{m})$ where $y(\mathbf{m})$ is defined.
- Neumann -The Neumann Boundary conditions are where we know the functions derivative in the direction normal to the boundary, i.e. $\frac{d}{d\hat{\nu}} f(\mathbf{m}) = \frac{d}{d\hat{\nu}} y(\mathbf{m})$
- Robin - The Robin Boundary conditions are where we know that some relationship exists between the function and its normal derivative, i.e. $\frac{d}{d\hat{\nu}} f(\mathbf{m}) = y(\mathbf{m}) - H f(\mathbf{m})$ where H is some constant.

Solving PDE's

One method for solving PDE's, which we use in this thesis, is to cast them in a variational form. To do this we will integrate them over the domain with some set of test

functions $\{v_k(\mathbf{r}); k = 1, 2, ...K\}$. If we start then from some function:

$$\mathcal{A}[f(\mathbf{r})] = 0 \tag{2.2}$$

where $f(\mathbf{r})$ is the function over the domain controlled by the operator $\mathcal{A}$, then we get an integral equation of the form:

$$\int_\Omega v(\mathbf{r})\mathcal{A}[f(\mathbf{r})]d\mathbf{r} = 0 \tag{2.3}$$

This finite dimensional representation of the space, with a solution whose error is orthogonal to the test space is referred to as a *weak formulation.*

Approximation Methods:

The usual method of approximating our model is to use an approximation such as equation 2.1. In which case equation 2.2 is not satisfied exactly but give rise to an error $\mathcal{A}[f(\mathbf{r})] = e$. We then have that $(v, e) = 0 \forall v \in V$ (the test space). It is clear then that in the general case we cannot provide a solution to the equations 2.2 or 2.3. We can make an approximation if we consider the test functions $v(\mathbf{r})$ as weighting functions, in other words:

$$v = w_j \quad \{j = 1, 2, ...M\} \quad where \quad M \leq N \tag{2.4}$$

This gives us equations of the form:

$$\int_\Omega w_j \mathcal{A}(\sum_{i=1}^{N} f_i u_i(\mathbf{r}))d\mathbf{r} = 0 \quad \{j = 1, 2, ...M\} \tag{2.5}$$

Here we have that $\mathcal{A}(\sum_{i=1}^{N} f_i u_i(\mathbf{r}))$ represents the residual or error of our approximation to the PDE . Hence we now have a weighted residual approximation to our problem. There are an variety of choices to the weighted functions, here we give a set of common choices:

Point Collocation: Here we chose a set of points $\{\mathbf{x}_j; j = 1, 2, ...J\}$ and set $w_j(\mathbf{r}) = \delta(\mathbf{r} - \mathbf{x}_j)$, where $\delta(\mathbf{r} - \mathbf{x}_j)$ is the Dirac Delta function. Here we are asserting that our residual is zero at the M points j.

Sub-domain collocation: Here we chose $w_j = 1$ in some sub-domain Ω_j, here effectively reducing the the error to zero in some sub-domain of the problem.

Galerkin Method: The Galerkin Method (also attributed to Bubnov), is where we set $M = N$ and $w_j = u_j$. It is used commonly as it has the interesting property of

usually leading to symmetric matrices. It also sets our residual zero at our N sample points.

2.1.4 Finite Difference Method

The Finite Difference Method (FDM) is another approach used to model Partial Differential Equations based on difference operators. A simple 1D example is given if we imagine trying to estimate $\frac{d}{dx}f(x)$ at a point i in a regularly sampled function, with spacing δ_x. We can easily see that we get :

$$\frac{d}{dx}f_i = \frac{f_{i+1} - f_{i-1}}{2\delta_x} \tag{2.6}$$

Using this approach we may discretise functions both spatially and temporally to model a given PDE. From such difference operators we can assemble a system of equations to represent our PDE, which can then be solved as a simple matrix equation.

More details of different FDM approaches can be found in most standard engineering maths texts, e.g. [109]. The drawback of the FDM is the limitation to structured meshing, i.e. to regular domains, as such it has certain disadvantages over the FEM. However in regular domains it is found that the structure of the FDM gives computationally more efficient methods.

2.1.5 Boundary Element Method

The Boundary Element Method (BEM) is a further way of solving PDE's. Here we discretise only the boundary $\partial\Omega$ of our domain (Ω) and solve the equation on this lower dimensional space. The BEM represents the PDE as an integral of the Green's function of the space between two points on the boundary. We may then establish a system of equations between all points on the boundary and solve for the function on the boundary. From here we can extrapolate the solution within the domain if necessary.

The BEM requires that our domain is homogeneous, however it can be extended to a multi-region model, allowing for piecewise homogeneous domains. To do this we must derive the BEM equations for each region and then equate the boundary conditions at the interface.

The BEM, initially, seems to be a better approach than the FEM, however it is constrained by certain limitations. First the domain must be homogeneous, although we can solve for multiple sub-regions. The second problem the BEM experiences is that

it produces a fully populated unsymmetric matrix which can be more difficult to solve than the sparse matrices produced by FEM approaches. Further details of the BEM can be found in [15]. we mention this here, for reference, as a BEM implementation supplied by Sikora[90] will be used for some numerical comparisons in chapter 7.

2.1.6 Spherical Harmonics and the Associated Legendre Polynomials

In this section we will outline spherical harmonics and their relevant properties. There are a selection of texts on these and the reader is encouraged to read these for more details (see for example [102]).

Spherical Harmonics

Spherical harmonics are the solution to Laplace's equation in spherical co-ordinates. We define the spherical harmonics as:

$$Y_l^m(\eta,\chi) = \sqrt{\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}}P_l^m(\cos(\eta))e^{im\chi} \tag{2.7}$$

where $P_l^m(\cos(\eta))$ is the associated Legendre polynomial (see section 2.1.6 below) with $m \in \{-l....0....l\}$ and the normalisation is chosen for convenience.

The spherical harmonics can be considered an analogy of the Fourier representation on a sphere. Increasing the order of the harmonics gradually increases the complexity of the functions which can be represented.

Associated Legendre Polynomials

The associated Legendre Polynomials are:

$$P_l^m(x) = (-1)^m(1-x^2)^{\frac{m}{2}}\frac{d^m}{dx^m}P_l(x) \tag{2.8}$$

where $P_l(x)$ is the standard Legendre Polynomial given by (in the Rodriguez representation):

$$P_l(x) = \frac{1}{2^l l!}\frac{d^l}{dx^l}(x^2-1)^l \tag{2.9}$$

Useful Properties of the Spherical Harmonics

Unit Vector in Spherical Components:

$$\hat{\mathbf{s}} = \begin{pmatrix} \sin\eta\cos\chi \\ \sin\eta\sin\chi \\ \cos\eta \end{pmatrix} = \sqrt{\frac{4\pi}{3}} \begin{pmatrix} \frac{1}{\sqrt{2}}(Y_{1,-1}(\hat{\mathbf{s}}) - Y_{1,1}(\hat{\mathbf{s}})) \\ \frac{1}{i\sqrt{2}}(Y_{1,-1}(\hat{\mathbf{s}}) + Y_{1,1}(\hat{\mathbf{s}})) \\ Y_{1,0}(\hat{\mathbf{s}}) \end{pmatrix}, \tag{2.10}$$

Addition Theorem:

$$P_l(\hat{\mathbf{x}}.\hat{\mathbf{y}}) = \frac{4\pi}{2l+1} \sum_{m=-l}^{l} Y^*_{l,m}(\hat{\mathbf{x}}) Y_{l,m}(\hat{\mathbf{y}}) \tag{2.11}$$

2.2 Meshing a Domain - Basic Principles

As we intend to model a PDE using a FEM approach it is clear we must have an understanding of mesh generation. In this section we will outline some of the classic approaches to mesh generation, to illustrate their relative strengths, and then discuss mesh refinement models. In this work we have looked at the differences between some of the refinement methods in terms of ease of use and producing more efficient techniques. It has become apparent that the key to an unstructured approach is a good basic mesh generation strategy, and this has led us to look into the approaches where structured meshes can be used to represent general shapes. First we must therefore say what we mean by a good quality mesh.

2.2.1 A brief note on quality meshes

There are two factors which are usually considered important to mesh generation. One is that the shape of elements does not bias our solution, in other words that we do not introduce large errors due to a poor spatial representation. The second is that we wish to minimise the size of our problem by reducing the number of nodes and computational cost to arrive at our solution.

In terms of the first question we know that skew or skinny elements can cause numerical problems in the FEM [9, 39]. In 2D this is a well understood problem and can be related to maximising the minimum aspect ratio of the mesh (the aspect ratio being the lowest ratio of the side to the height of the triangle). In 3D however it is difficult to ascertain an exact quality measure in such terms, one common approach from the literature is the ratio of in-sphere to out-sphere of the tetrahedron normalised

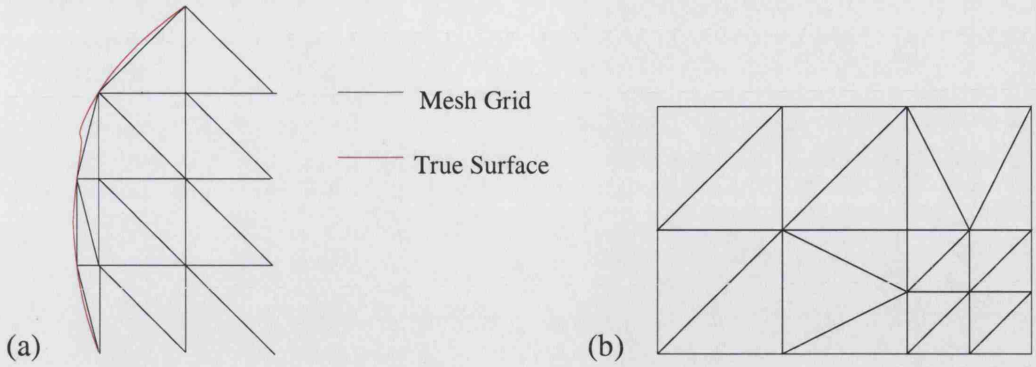


Figure 2.3: (a) A diagram illustrating the curved surface griding for a quad-tree. (b) A diagram illustrating a typical quad-tree localised refinement.

by the ratio for a platonic tetrahedra. Currently we examine quality by another easier measure, of worst height to side ratio, i.e. the largest ratio of a side to the height of the element using this as the base - this gives an approximate idea of skew.

2.2.2 Classic Volume Meshing Methods

In this section we will outline the four most common meshing strategies. There are several readily available meshing tools which we have used in our work, it has been a question of identifying the best tool for our problem.

Quad/Oct-Tree

Quad or Oct-Tree meshing is a simple regular grid approach. It fills the volume with regular sized pixels or voxels, which are then converted to triangles/tetrahedra (further details can be found in any standard meshing text e.g. [98]). There are two additional features which are important to note for the tree approach to mesh generation as they are what make it less useful for our purposes.

First and foremost the fitting to an irregular domain is done by extending the contained pixels/voxels to the edge and then creating triangles from these (this approach is illustrated in figure 2.3(a)). The second, less important factor is how local refinement is achieved. This is done by allowing the grid size to be reduced locally, normally allowing only one step size between neighbouring elements (see figure 2.3(b)).

The two features give the following problems:

- The rate of local adaptation is strictly limited if we wish to avoid skew elements.

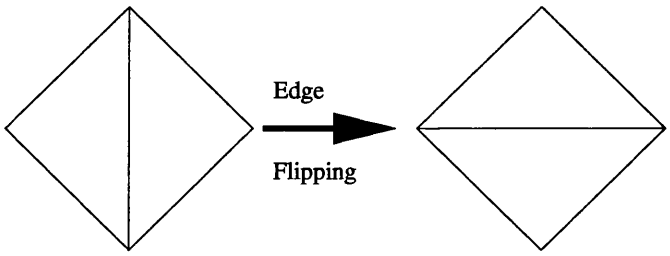


Figure 2.4: A figure illustrating the process of edge flipping used in Delauney triangulation methods

- The curvature of the surface will strongly effect the skew elements near the surface.

These combine to give us a poor quality control in our meshes, although it is worth noting that this is easily the most time efficient way to generate a mesh as it is simple and fast. Quad/Oct-tree generators are also readily available, e.g. the 3D algorithm for QMG [100].

Delauney Triangulation

Probably the best known mesh generation algorithm is the Delauney triangulation method (for example see [53, 21, 98]), a detailed review and possibly the most commonly used model can be found in [79]. The Delauney triangulation in 2D is a well known algorithm and relies on minimising the skew elements of the mesh. The problem is slightly more complex in 3D as the error metric is less well defined and also because the method relies on flipping elements on a given edge if their aspect can be improved - in 2D only 1 operation is possible, whereas multiple operations are allowed in 3D for flipping on a shared face (the 2D equivalent of the edge). Edge flipping is the process where you examine both triangles on a given edge and if by exchanging the connection you improve the quality of the element then you do so this is illustrated in figure 2.4.

Advancing Front Technique

The Advancing Front Technique (AFT), generates a triangulation by gradually growing a volume mesh out from the surface. It is not as stable as Delauney and can lead to numbers of low quality elements, but is usually faster and easier to implement. Details of the AFT can be found in the standard texts (e.g. [98]) or for example in [82] with an associated package and code available at [81].

Essentially the technique works by dividing an $(n-1)$ Dimensional surface[1] into a list of $(n-1)$Dimensional sides in a front-line. From this front-line you select a side to grow into an element, the old side is removed from the front and the other sides of the new element are added - this process proceeds until the mesh is complete. The problems with the AFT are in choosing the side to grow and in the setting up of the new element, this must be done within the parameters of quality and size of the desired mesh, but in the end must also lead to a filled space.

Sphere-Packing

Sphere packing is probably the hardest of the mesh codes to implement, and the least provable in terms of mesh quality. However it is possibly intuitively the best way to obtain good quality meshes. Results do in fact show that it does tend to produce meshes of quality, but there is no mathematical bounding proof of this, as there is for Delauney (see [79]).

There are various approaches to sphere-packing, the most detailed description can be found in Shimada's PhD thesis [88], where a technique called bubble meshing is described. The idea of sphere packing is to surround each point in a mesh with an electrostatic sphere, the forces on these spheres can be used to vibrate the points into an even distribution by use of standard force equations. Sphere packing can be made more dynamic by introducing insertion and deletion rules in areas of high/low node density and by using node-spacing functions (NSF's) to give each node a different force, thus producing anisotropic, or adapted meshes.

2.2.3 Mesh Refinement Strategies

Mesh refinement is an important question in mesh generation. Sometimes we may wish to refine a mesh as the solution it generates to our mathematical problem is poor. Such refinements can be done globally, but often we may find it better to perform local refinements, thus only making the problem more complex where necessary. There are three basic strategies for mesh refinement:

- h-refinement: Increasing the number of elements in the domain (see figure 2.5a).
- p-refinement: Increasing the order of the elements in the domain (see figure

[1] where n is the dimensionality of the mesh.

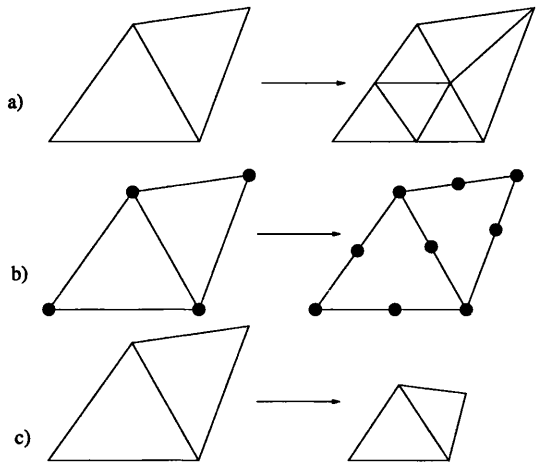


Figure 2.5: A diagram illustrating the various refinement methods available in FEM, a) h-refinement, b) p-refinement and c) r-refinement.

2.5b).

- r-refinement: Redistributing the original elements to better sample the function (see figure 2.5c).

Each of these approaches has its merits, it is noticeable however that the first two tend to be used for global refinement and the latter two for local refinement strategies. Strictly the first h-refinement can be used for localised refinements but doing this creates more skinny triangles than before and therefore reduces mesh quality (see the right hand triangle in figure 2.5a). In our work we have implemented two refinement strategies we will discuss later, the idea has been to focus on localised refinement so we have implemented a Laplacian smoothing approach based on [14] and a local p-refinement method.

2.2.4 Existing Software

Various packages have been tried for the generation of meshes. Most types of algorithm (except sphere-packing) are available as free-ware packages, a good survey can be found at Steve Owen's meshing research corner [65]. From those available several have been tried, these have been assessed on usability, and quality of meshes produced. QMG was the first package used and whilst it produces reasonable 2D meshes in 3D it uses the Oct-Tree approach, using large elements away from the surface, which proves to require very high density meshes to provide anything approaching a stable mesh. Un-

fortunately no sphere-packing algorithms are currently freely made available, so whilst they may have provided some advantages these approaches have not been adopted as construction of a mesh-generation package is outside the scope of this thesis.

Other packages tried have included two Delauney based codes GRUMMP [64] and Tetgen [89], however both of these have a difficult command line interface and require initial surface triangulations as a starting point. Currently we use NETGEN [81], an advancing front package, as this has a relatively easy GUI front end and allows for meshes to be constructed using Computational Solid Geometry (CSG). NETGEN also allows for local points to be more highly refined and also for local refinement based on more complex CSG objects - e.g. insertion of a source plane to the model, constraining the mesh to be finer in this plane.

2.3 Summary

We have outlined the basic mathematical and geometric tools required to complete this work. We have described the principles of approximating and solving PDE's using FDM, FEM and BEM. We have introduced the spherical harmonics used to handle the directional component of the RTE and to reduce it to the DA. Finally we have introduced the standard mesh generation techniques, and given an explanation of the methods adopted in this work.

Chapter 3

Transport Theory - Atmospheric, Neutron and Optical Transport

In this chapter we will first introduce the Radiative Transfer Equation (RTE - equation 1.1), in terms of how it is assembled from its components (section 3.1). We will then go on to outline some of the classical approaches to solving this equation in the transport literature. We will show how approaches to solving the full RTE have developed over the years (sections 3.2 and 3.3), illustrating their strengths and weaknesses. We shall also introduce the concept of using approximation where appropriate to reduce the computational costs of solving the equation (section 3.4), and show how this limits the domain we can solve. Finally, in section 3.5, we will give an overview of the use of these methods in Optical Tomography.

3.1 The Radiative Transfer Equation

In this section we will outline the basics of transport theory in terms of the Radiative Transfer Equation. We explain the various physical components involved in the transport process and then combine them to form the balance equation referred to as the RTE. The equation is a description of the transport of particles through space and has been used in, for example, atmospherics [47, 48, 20], neutron transport [1, 57] and light transport - both for graphics [7, 51, 93, 95] and in medical imaging (optical tomography) [3, 26, 43, 52, 66], other more general discussions can be found in [19, 30]. We will then illustrate the transformation of the equation to the even/odd parity form, a common tool used when considering the RTE.

3.1.1 The components of the Radiative Transfer Equation

Here we outline the various components of particle transport which build to form the RTE. In the next section we will show how these components combine as a balance equation to form the RTE. The components of the RTE determine the flux of particles in the medium, $\phi(\mathbf{r}, \hat{\mathbf{s}}, t)$, i.e. the density of photons travelling in direction $\hat{\mathbf{s}}$, in the volume around $\mathbf{r}$ at time t, photons mm^3 sr^{-1} ps^{-1}.

Change of particle flow in time

If our sources of particles are time dependent , then there will be some time-variant component of the particle flow. This can be expressed simply as $\frac{d\phi(\mathbf{r},\hat{\mathbf{s}},t)}{dt}$, the rate of change of the flux in time. This term will form the basis of the time-dependent form of the RTE as it provides the start for the balance equation.

Through Transport or Leakage

Some particles in a system will pass straight through volumes in space remaining unaffected by the presence of scatterers/absorbers. These particles form the through transport, leakage or advection component of the RTE, their behaviour is illustrated in figure 3.1.

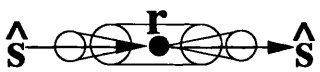


Figure 3.1: A figure illustrating the through transport in direction $\hat{\mathbf{s}}$ at a point $\mathbf{r}$.

The term which gives the through transport is derived as follows, if we consider a small volume around point $\mathbf{r}$ as illustrated in figure 3.2:

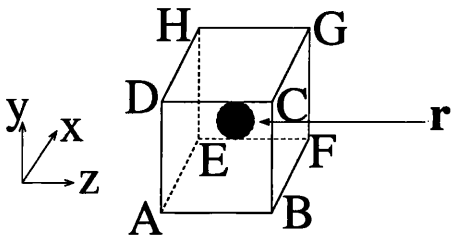


Figure 3.2: A figure illustrating a differential volume being considered around point $\mathbf{r}$.

We can now consider the leakage across each face in the same sense as the normal

to that face, for face ABCD, this gives us:

$$\text{Leakage}_{ABCD} \approx \hat{\mathbf{s}} \cdot \mathbf{i}\phi(\mathbf{r} + \frac{dx}{2}, \hat{\mathbf{s}}, t)$$

and for face EFGH:

$$\text{Leakage}_{EFGH} \approx \hat{\mathbf{s}} \cdot -\mathbf{i}\phi(\mathbf{r} - \frac{dx}{2}, \hat{\mathbf{s}}, t)$$

where here dx represents the size of our differential volume in this direction and $\mathbf{i}$ is the standard unit vector in the x-direction. These combine to give us the Leakage in one axis (or dimension) as:

$$\text{Leakage}_i \approx \hat{\mathbf{s}} \cdot \mathbf{i}\frac{d}{dx}\phi(\mathbf{r}, \hat{\mathbf{s}}, t)$$

Summing this over each of the axes gives us the total through transport term for $\mathbf{r}$ as:

$$\hat{\mathbf{s}} \cdot \nabla\phi(\mathbf{r}, \hat{\mathbf{s}}, t) \tag{3.1}$$

Transport Attenuation

The transport attenuation term, given by the transport coefficient μ_t gives the reduction in flow caused by the transport losses. There are two types of transport loss to consider, absorption by the medium (regulated by μ_a, the absorption coefficient) and out-scatter (regulated by μ_s, the scattering coefficient) where the particle is scattered out of the direction, $\hat{\mathbf{s}}$, being considered. Both these processes are stochastic and will depend on the medium, they can be described in more physical terms, but for the purposes of this thesis it is enough to consider them as properties of the medium. These two effects are illustrated in figure 3.3, we note that $\mu_t = \mu_s + \mu_a$.

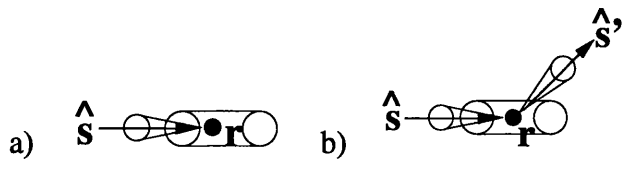


Figure 3.3: A figure illustrating the transport loss in direction $\hat{\mathbf{s}}$ at a point $\mathbf{r}$, due to a) absorption and b) out-scatter

This gives us a reduction term as:

$$\mu_t(\mathbf{r})\phi(\mathbf{r}, \hat{\mathbf{s}}, t) \tag{3.2}$$

In-Scattering

The concept of loss due to out-scatter implies the need for another term, the in-scattering term. This is the process whereby light scatters into the direction of interest $\hat{\mathbf{s}}$ from some other direction $\hat{\mathbf{s}}'$, and is illustrated in figure 3.4.

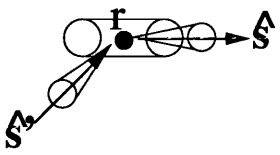


Figure 3.4: A figure illustrating the gain of particles due to in-scattering to direction $\hat{\mathbf{s}}$ from direction $\hat{\mathbf{s}}'$ at a point $\mathbf{r}$.

The in-scatter term is governed by two components, the scattering coefficient, μ_s, and the scattering probability distribution $\Theta(\mathbf{r}, \hat{\mathbf{s}}, \hat{\mathbf{s}}')$. $\Theta(\mathbf{r}, \hat{\mathbf{s}}, \hat{\mathbf{s}}')$ represents the probability that a photon travelling in direction $\hat{\mathbf{s}}'$ will scatter into direction $\hat{\mathbf{s}}$, we require that $\int_{S^{n-1}} \Theta(\mathbf{r}, \hat{\mathbf{s}}, \hat{\mathbf{s}}')d\hat{\mathbf{s}}' = 1$ and that $0 <= \Theta(\mathbf{r}, \hat{\mathbf{s}}, \hat{\mathbf{s}}') <= 1$. To find the amount of in-scattering then we must integrate over the set of possible directions S^{n-1} in $\hat{\mathbf{s}}'$ to give our contribution to the term in $\hat{\mathbf{s}}$. This gives us the in-scatter term as:

$$\mu_s(\mathbf{r}) \int_{S^{n-1}} \Theta(\mathbf{r}, \hat{\mathbf{s}}, \hat{\mathbf{s}}')\phi(\mathbf{r}, \hat{\mathbf{s}}', t)d\hat{\mathbf{s}}' \tag{3.3}$$

Source Term

The final term we must consider is the concept of particle sources. It is described directly as its own function, $q(\mathbf{r}, \hat{\mathbf{s}}, t)$, and is illustrated in figure 3.5.

Figure 3.5: A figure illustrating the emission of particles in direction $\hat{\mathbf{s}}$ from a point $\mathbf{r}$.

3.1.2 Assembling the balance equation

The transport problem then can be considered as a balance problem of the factors described in section 3.1.1. This will give us the rate of change of the photon flux with time as the sum of the additive elements less the sum of the subtractive elements. The additive components are the particle source and in-scatter terms, whilst the subtractive elements are the transport and through transport terms.

The RTE, is obtained by combining the rate of change of flux (modified for the particle speed) with the terms given in equations 3.1, 3.2, 3.3 and the source term. This gives us :

$$\frac{1}{c}\frac{d\phi(\mathbf{r},\hat{\mathbf{s}},t)}{dt}+\hat{\mathbf{s}}.\nabla\phi(\mathbf{r},\hat{\mathbf{s}},t)+\mu_t(\mathbf{r})\phi(\mathbf{r},\hat{\mathbf{s}},t)=\mu_s(\mathbf{r})\int\Theta(\mathbf{r},\hat{\mathbf{s}},\hat{\mathbf{s}}')\phi(\mathbf{r},\hat{\mathbf{s}}',t)d\hat{\mathbf{s}}'+q(\mathbf{r},\hat{\mathbf{s}},t), \tag{3.4}$$

This is the standard form of the Radiative Transfer Equation. It describes the transport of particles through media with scattering and or absorption effects. It can be reduced to simpler forms for specific cases where we can make simplifications based on the media's properties, some examples of this and numerical solutions to these will be given in the following sections.

3.1.3 Parity Form of the RTE

A common tool when solving the RTE is to split it into its even and odd parity form, the parity forms are where in the even case the flux is symmetric and the odd parity is where the flux is anti-symmetric. It can sometimes be easier to solve for one of these and then to derive the other, and hence the full solution from these two terms.

The idea of the Parity forms of the RTE is to split the flux into its even and odd parity form, these are given by:

$$\text{Even Parity Flux} : \quad \phi_+(\mathbf{r},\hat{\mathbf{s}},t)=\tfrac{1}{2}\left[\phi(\mathbf{r},\hat{\mathbf{s}},t)+\phi(\mathbf{r},-\hat{\mathbf{s}},t)\right] \tag{3.5}$$

$$\text{Odd Parity Flux} : \quad \phi_-(\mathbf{r},\hat{\mathbf{s}},t)=\tfrac{1}{2}\left[\phi(\mathbf{r},\hat{\mathbf{s}},t)-\phi(\mathbf{r},-\hat{\mathbf{s}},t)\right] \tag{3.6}$$

we will also need that:

$$\Theta_+(\mathbf{r},\hat{\mathbf{s}},\hat{\mathbf{s}}') = \frac{1}{2}\left[\Theta(\mathbf{r},\hat{\mathbf{s}},\hat{\mathbf{s}}')+\Theta(\mathbf{r},-\hat{\mathbf{s}},\hat{\mathbf{s}}')\right] \tag{3.7}$$

$$\Theta_-(\mathbf{r},\hat{\mathbf{s}},\hat{\mathbf{s}}') = \frac{1}{2}\left[\Theta(\mathbf{r},\hat{\mathbf{s}},\hat{\mathbf{s}}')-\Theta(\mathbf{r},-\hat{\mathbf{s}},\hat{\mathbf{s}}')\right] \tag{3.8}$$

$$\Theta(\mathbf{r},\hat{\mathbf{s}},\hat{\mathbf{s}}') = \Theta_+(\mathbf{r},\hat{\mathbf{s}},\hat{\mathbf{s}}')+\Theta_-(\mathbf{r},\hat{\mathbf{s}},\hat{\mathbf{s}}') \tag{3.9}$$

Starting from equation 3.4 in the $-\hat{\mathbf{s}}$ direction, we have:

$$\begin{aligned}\frac{1}{c}\frac{d\phi(\mathbf{r},-\hat{\mathbf{s}},t)}{dt} &- \hat{\mathbf{s}}.\nabla\phi(\mathbf{r},-\hat{\mathbf{s}},t)+\mu_t(\mathbf{r})\phi(\mathbf{r},-\hat{\mathbf{s}},t)\\ &= \mu_s(\mathbf{r})\int\Theta(\mathbf{r},-\hat{\mathbf{s}},\hat{\mathbf{s}}')\phi(\mathbf{r},\hat{\mathbf{s}}',t)d\hat{\mathbf{s}}'+q(\mathbf{r},-\hat{\mathbf{s}},t),\end{aligned} \tag{3.10}$$

Adding equations 3.4 and 3.10,and using equation 3.7, we get:

$$\frac{1}{c}\frac{d\phi_+(\mathbf{r},\hat{\mathbf{s}},t)}{dt} + \hat{\mathbf{s}}.\nabla\phi_-(\mathbf{r},\hat{\mathbf{s}},t) + \mu_t(\mathbf{r})\phi_+(\mathbf{r},\hat{\mathbf{s}},t) = \mu_s(\mathbf{r})\int \Theta_+(\mathbf{r},\hat{\mathbf{s}},\hat{\mathbf{s}}')\phi(\mathbf{r},\hat{\mathbf{s}}',t)d\hat{\mathbf{s}}' + q_+(\mathbf{r},\hat{\mathbf{s}},t), \quad (3.11)$$

by splitting $\phi(\mathbf{r},\hat{\mathbf{s}}',t)$ into $\phi_+(\mathbf{r},\hat{\mathbf{s}}',t)+\phi_-(\mathbf{r},\hat{\mathbf{s}}',t)$ and remembering that odd functions integrate to zero over a sphere, we get:

$$\frac{1}{c}\frac{d\phi_+(\mathbf{r},\hat{\mathbf{s}},t)}{dt} + \hat{\mathbf{s}}.\nabla\phi_-(\mathbf{r},\hat{\mathbf{s}},t) + \mu_t(\mathbf{r})\phi_+(\mathbf{r},\hat{\mathbf{s}},t) = \mu_s(\mathbf{r})\int \Theta_+(\mathbf{r},\hat{\mathbf{s}},\hat{\mathbf{s}}')\phi_+(\mathbf{r},\hat{\mathbf{s}}',t)d\hat{\mathbf{s}}' + q_+(\mathbf{r},\hat{\mathbf{s}},t), \quad (3.12)$$

In a similar way subtracting equations 3.4 and 3.10, we get:

$$\frac{1}{c}\frac{d\phi_-(\mathbf{r},\hat{\mathbf{s}},t)}{dt} + \hat{\mathbf{s}}.\nabla\phi_+(\mathbf{r},\hat{\mathbf{s}},t) + \mu_t(\mathbf{r})\phi_-(\mathbf{r},\hat{\mathbf{s}},t) = \mu_s(\mathbf{r})\int \Theta_-(\mathbf{r},\hat{\mathbf{s}},\hat{\mathbf{s}}')\phi_-(\mathbf{r},\hat{\mathbf{s}}',t)d\hat{\mathbf{s}}' + q_-(\mathbf{r},\hat{\mathbf{s}},t). \quad (3.13)$$

Equations 3.12 and 3.13 form the basis of the parity form of the RTE. The parity form is given by by combing equations 3.12 and 3.13 and eliminating either ϕ_+ or ϕ_-.

3.2 Numerically Solving the Full RTE

In this section we will discuss the three most common numerical approaches to solving the full transport equation, the Discrete Ordinates Method (DOM), the Phase-Space Finite Element Method (PSFEM) and the P_N method. We examine these in this order partially as a historical sequence but also as this naturally leads us to approximation approaches at the end. The common denominator in these methods is the large computational expense in generating full solutions, it can be seen that in certain environments less expense will produce a better model, but in all cases as we reduce the scattering coefficient to zero, we increase our computational cost.

Full transport equations are often used in applications such as astrophysics as we are dealing with varying media with large ranges of properties. They have also been employed in fields where we are dealing with largely diffusing media, such as reactor physics and optics, when small regions of the domain become non-diffusive. Here we will outline the most common of these approaches, pointing to their individual weaknesses. We will show how these models build upon each other, but we will also

indicate why they can become expensive in non-scattering spaces. This is even if we ignore the fact that they are usually full domain models, i.e. they solve area's with a high computational cost where a simpler model would suffice. This is one of the trends that took us to looking at hybrid approximations to solve our particular transport problem.

3.2.1 Discrete Ordinates

The Discrete Ordinate Method (DOM), historically referred to as the S_N method, is one of the older approaches to solving the full transport equation. It is commonly used in Neutron Transport, see for example [1, 57], and a full description can be found in most transport theory texts, see for example [30, 57]. It is also used in the field of light transport, see for example [12, 26, 27, 43].

The DOM first reduces the directional component of the RTE to a set of discrete directions. Next the spatial domain is discretised using a standard finite difference regime. This allows us to generate the full solution by an iterative process of sweeping through the domain using the directional components. We can, as has been shown in [12, 26, 27, 43], use this approach to produce solutions to the transport equation very readily. It is however computationally expensive, particularly if we have low-scattering regions. This is due to the use of discrete directions, which produce something referred to as the ray effect which we will explain in the next section. The potential problems with this in Optical Tomography were highlighted in 2D in [4].

The Ray-Effect

The ray effect is a well known defect with the DOM. It arises from the fact that we consider only discrete directions for the flux. It is easiest to comprehend if we consider a 2D case of a region with no scattering and, say, 8 ordinate directions. If we place an isotropic source at the centre of such a region, we should get light to all parts of our domain, however if we simply consider the problem we can draw in regions where there will be no light, see figure 3.6. Here we see that light, when exclusively transmitted along the discrete rays in a non-scattering space, fails to correctly illuminate the scene. A figure such as figure 3.6 is easily generated, we can see the regions the light does not reach simply by examining which elements in the grid have no connectivity to the nodes/elements traced by the discrete rays.

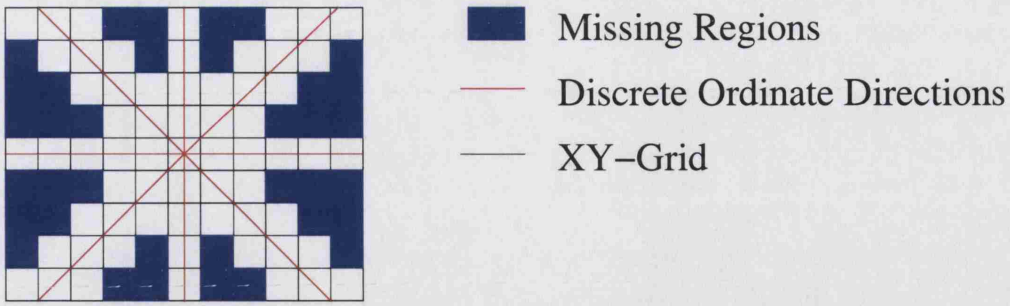


Figure 3.6: A diagram illustrating the regions where light does and does not reach from an isotropic source in a 2D non-scattering domain DOM solution with 8 ordinates - the ray effect.

The ray effect is considered in more detail by Lathrop [54]. Here Lathrop shows that in complex domains with scatter, the ray effect can cause intuitively incorrect numerical results, also that it may just cause numerical error whilst appearing correct. However these are not largely considerations we need to concern ourselves with, as it is clear that in non-scattering regions the DOM can have serious problems, as these form part of our domain we must find a remedy for them or consider other possible solutions.

Solutions to the Ray-Effect

There are many potential remedies for the ray-effect, discussed in [55, 58]. There are of course simple suggestions such as extending the number of ordinates till a stable/correct solution is obtained, but as Lathrop showed [54], stable does not necessarily imply correct in this case and as has been shown in [4] the comparative cost of this to other techniques is quite prohibitive.

Max [58] suggests we use a bin solution as opposed to an ordinate solution. The idea is instead of having a discrete set of directions which may exclude parts of the domain incorrectly, we should effectively use a set of zero order basis functions between ordinate directions. The idea has the benefit of not really increasing the cost of our sweeps in the DOM in terms of directions (we have the same number), but does increase the complexity of the calculations. Max notes that in order to keep an even distribution of energy we must be careful of our weights w_n. Such an approach though is part way to a more general model, the Phase-Space Finite Element Method, discussed

below in section 3.2.2, it is suggested that this is a more useful form of this solution due to its generality.

Lathrop [55] proposes a variety of solutions to speed up the convergence of the DOM under the ray effect. These however do not solve the underlying problem that this convergence may in some cases be incorrect [54]. He does suggest one method which may mitigate the ray effect completely, an approach he calls spherical harmonic-like. In doing so he shows that an equivalence can be obtained in the S_N DOM equations to an order of $N-3$ in the P_N method discussed below (section 3.2.3). This approach seems interesting but is as costly as a P_N solution whilst obtaining a lower degree of accuracy.

Usefulness of the DOM

As we have discussed above the primary disadvantage of the DOM is the ray-effect. This effect is most exaggerated in regions of low or no scatter, clearly this means that whilst the DOM may be a suitable model for highly scattering regions, its usefulness decreases in the presence of reduced scattering. We note here that as we are trying to develop a model for non-scattering spaces it may be unwise to use a DOM based approach. In fact in experiments in 2D [4] show that the order of discrete ordinates required rises rapidly in the presence of a clear layer. We surmise then that a more complete directional basis is required to handle the problem and move on to consider other approaches which fulfil this criteria.

3.2.2 Phase-Space Finite Element Method

The Phase-Space Finite Element Method (PSFEM) is an approach which allows us to use generic geometries and generic descriptions of the angular component of the flux. We note here that the phase referred to in the name of this method is the angular component $\hat{\mathbf{s}}$ of the flux, it is a historic reference to the term used in atmospherics for the angular component. It avoids the Ray Effect by using basis functions to describe not only the spatial, but also the angular variation of light in the domain. We can consider it as a two stage process, first we express the spatial, $\mathbf{r}$, component of ϕ in terms of a set of basis functions and then we perform a second discretisation in angle, $\hat{\mathbf{s}}$. We may consider it as representing $\phi(\mathbf{r},\hat{\mathbf{s}})$ as $\phi_{i,x}u_i(\mathbf{r})v_x(\hat{\mathbf{s}})$. It is presented, in detail, in [59, 60] in first one and then two dimensions.

We may consider Max's approach [58] as the use of zero order basis functions to represent the angular component. The PSFEM has the advantage of enabling us to chose our basis and also as it is a FEM model we may allocate them more freely, rather than in the even axial configuration of the DOM.

We have then a system of equations based in spatial and angular domains, which again have a similar order to the DOM. The advantage here is that we now have a continuous angular component which should mitigate the ray-effect, with at least C^0 continuity (given that we use at least first order elements in the angular domain). However as [54] suggests this may not be enough to truly mitigate the ray-effect, suggesting that we may be better to use a method using a more natural spherical harmonic basis for the angular component such as the P_N method outlined below.

3.2.3 P_N Model

Another approach to modelling the full RTE is to use a spherical harmonic basis for representing the phase component, see [3, 8, 36, 37]. This approach is referred to as the P_N Method, it is given by the spherical harmonic expansion of equation 3.4 in $\hat{\mathbf{s}}$ and $\hat{\mathbf{s}}'$. The first stage is to express the quantities in 3.4 in their harmonic expansions:

$$\phi(\mathbf{r},\hat{\mathbf{s}},t) = \sum_{l}^{\infty}\sum_{m=-l}^{l}\left(\frac{2l+1}{4\pi}\right)^{\frac{1}{2}}\Psi_{l,m}(\mathbf{r},t)Y_{l,m}(\hat{\mathbf{s}}) \tag{3.14}$$

$$q(\mathbf{r},\hat{\mathbf{s}},t) = \sum_{l}^{\infty}\sum_{m=-l}^{l}\left(\frac{2l+1}{4\pi}\right)^{\frac{1}{2}}q_{l,m}(\mathbf{r},t)Y_{l,m}(\hat{\mathbf{s}}) \tag{3.15}$$

where using (η,χ) as the spherical coordinates of $\hat{\mathbf{s}} \in S^2$ we can then use the definition of the spherical harmonics as given in section 2.1.6.

Further if we use the addition theorem (equation 2.11) and assume that $\Theta(\mathbf{r},\hat{\mathbf{s}},\hat{\mathbf{s}}')$ is in direction terms only a function of the angle between $\hat{\mathbf{s}}$ and $\hat{\mathbf{s}}'$, i.e. $\Theta(\mathbf{r},\hat{\mathbf{s}},\hat{\mathbf{s}}') = \Theta(\mathbf{r},\hat{\mathbf{s}}\cdot\hat{\mathbf{s}}')$, we get:

$$\begin{aligned}\Theta(\mathbf{r},\hat{\mathbf{s}}.\hat{\mathbf{s}}') &= \sum_{l}^{\infty}\left(\frac{2l+1}{4\pi}\right)\Theta_l(\mathbf{r})P_l(\hat{\mathbf{s}}\cdot\hat{\mathbf{s}}')) \\ &= \sum_{l}^{\infty}\sum_{m=-l}^{l}\Theta_l(\mathbf{r})Y^*_{l,m}(\hat{\mathbf{s}}')Y_{l,m}(\hat{\mathbf{s}})\end{aligned} \tag{3.16}$$

where $Y^*_{l,m}(\hat{\mathbf{s}}')$ is the complex conjugate of $Y_{l,m}(\hat{\mathbf{s}})$. Now given the expression of a unit vector in terms of its spherical components (see equation 2.10), we can express

equation 3.4 in terms of it's spherical harmonics as:

$$\begin{aligned}\left(\frac{1}{c}\frac{\partial}{\partial t}+\mu_t(\mathbf{r})\right) &\sum_{l}^{\infty}\sum_{m=-l}^{l}\left(\tfrac{2l+1}{4\pi}\right)^{\frac{1}{2}}\Psi_{l,m}(\mathbf{r},t)Y_{l,m}(\hat{\mathbf{s}})+\sum_{l}^{\infty}\sum_{m=-l}^{l}\left(\tfrac{2l+1}{4\pi}\right)^{\frac{1}{2}}\hat{\mathbf{s}}\cdot\nabla\Psi_{l,m}(\mathbf{r},t)Y_{l,m}(\hat{\mathbf{s}})\\ =\ & \mu_s(\mathbf{r})\int_{S^{n-1}}\left(\sum_{l}^{\infty}\sum_{m=-l}^{l}\Theta_l(\mathbf{r})Y^*_{l,m}(\hat{\mathbf{s}}')Y_{l,m}(\hat{\mathbf{s}})\right)\\ &\times\left(\sum_{l}^{\infty}\sum_{m=-l}^{l}\left(\tfrac{2l+1}{4\pi}\right)^{\frac{1}{2}}\Psi_{l,m}(\mathbf{r},t)Y_{l,m}(\hat{\mathbf{s}}')\right)d\hat{\mathbf{s}}'\\ &+\sum_{l}^{\infty}\sum_{m=-l}^{l}\left(\tfrac{2l+1}{4\pi}\right)^{\frac{1}{2}}q_{l,m}(\mathbf{r},t)Y_{l,m}(\hat{\mathbf{s}})\end{aligned}\tag{3.17}$$

Now we can de-couple this equation by taking the inner product with $Y^*_{l,m}(\hat{\mathbf{s}})$, and by using the following relations:

$$\begin{aligned}\cos(\eta)Y_{l,m} &= \left(\frac{(l+m)(l-m)}{(2l+1)(2l-1)}\right)^{\frac{1}{2}}Y_{l-1,m}\\ &+\left(\frac{(l+m+1)(l-m+1)}{(2l+1)(2l+3)}\right)^{\frac{1}{2}}Y_{l+1,m}\end{aligned}\tag{3.18}$$

$$\begin{aligned}\sin(\eta)e^{i\chi}Y_{l,m} &= \left(\frac{(l-m)(l-m-1)}{(2l+1)(2l-1)}\right)^{\frac{1}{2}}Y_{l-1,m+1}\\ &-\left(\frac{(l+m+1)(l+m+2)}{(2l+1)(2l+3)}\right)^{\frac{1}{2}}Y_{l+1,m+1}\end{aligned}\tag{3.19}$$

$$\begin{aligned}\sin(\eta)e^{-i\chi}Y_{l,m} &= \left(\frac{(l+m)(l+m-1)}{(2l+1)(2l-1)}\right)^{\frac{1}{2}}Y_{l-1,m-1}\\ &-\left(\frac{(l-m+1)(l-m+2)}{(2l+1)(2l+3)}\right)^{\frac{1}{2}}Y_{l+1,m-1}\end{aligned}\tag{3.20}$$

which gives us (after some algebra) the infinite series of equations:

$$\begin{aligned}&\left(\tfrac{1}{c}\tfrac{\partial}{\partial t}+\mu_t(\mathbf{r})\right)\Psi_{l,m}(\mathbf{r},t)\\ +\ &\tfrac{1}{2l+1}\Big(\Big[\tfrac{\partial}{\partial z}\Big((l-m+1)^{\frac{1}{2}}(l+m+1)^{\frac{1}{2}}\Psi_{l+1,m}(\mathbf{r},t)\\ &+(l-m)^{\frac{1}{2}}(l+m)^{\frac{1}{2}}\Psi_{l-1,m}(\mathbf{r},t)\Big)\Big]\\ -\ &\Big[\tfrac{1}{2}\Big(\tfrac{\partial}{\partial x}-i\tfrac{\partial}{\partial y}\Big)\Big((l+m)^{\frac{1}{2}}(l+m-1)^{\frac{1}{2}}\Psi_{l-1,m-1}(\mathbf{r},t)\\ &-(l-m+2)^{\frac{1}{2}}(l-m+1)^{\frac{1}{2}}\Psi_{l+1,m-1}(\mathbf{r},t)\Big)\Big]\\ -\ &\Big[\tfrac{1}{2}\Big(\tfrac{\partial}{\partial x}+i\tfrac{\partial}{\partial y}\Big)\Big((l+m+1)^{\frac{1}{2}}(l+m+2)^{\frac{1}{2}}\Psi_{l+1,m+1}(\mathbf{r},t)\\ &-(l-m)^{\frac{1}{2}}(l-m-1)^{\frac{1}{2}}\Psi_{l-1,m+1}(\mathbf{r},t)\Big)\Big]\Big)\\ &=\mu_s(\mathbf{r})\Theta_l(\mathbf{r})\Psi_{l,m}(\mathbf{r},t)+q_{l,m}(\mathbf{r},t)\end{aligned}\tag{3.21}$$

Equation 3.21 forms the P_N Method, we get the individual equations for any P_N by truncating the series to the limit $l <= N$ giving us $(N+1)^2$ equations.

The P_N model for solving the Transport equation may be considered as an approximation, by truncation of the series. It may also be considered that such truncation can be performed to an error limit, making it a full transport model. One classical use of the P_N approach is outlined in section 3.4.1 as it forms the basis of the Diffusion Approximation. Other approaches to using the P_N model are given in [8, 36, 37, 105].

3.3 Stochastic Models for the RTE - Monte-Carlo Simulation

One classical approach to solve the transport problem is to treat it as a stochastic problem and solve it with a Monte-Carlo or random walk method. The approach here is to plot the path of individual photons through a medium and sum up their effect until a stable result is obtained.

Given a domains scattering and absorption characteristics a model is built, which allows us to calculate the path of a photon through the domain. The following algorithm for photons can then be used repeatedly until a stable model has been built for photon behaviour:

- Launch photon.
- Given the transport coefficient (probability of event) in the domain, calculate distance to first event.
- Loop until photon destroyed or leaves medium:
  - At event determine whether absorption or scatter
    - if absorption end loop
    - if scatter calculate next event direction and distance
  - Check against maximum photon path length - if reached end loop
  - Check photon still in domain - if not store data at exit point and end loop
- Read out photon density at surface.

This forms an outline of Monte-Carlo simulation and suffices to understand the principle of the method. Obviously if there are internal boundaries in the domain where

properties change this algorithm becomes more complex, but the principle is the same. In this work we only use Monte-Carlo simulations for the purposes of numerical validation of models. This is purely as even with geometrically simple models, e.g. spheres, where code can exploit symmetry to accelerate the process, such models can be extremely time consuming. These models are used in all transport fields, partially as validation but also where other numerical solutions prove as cumbersome. Often, the maximum photon path length is added to the code in order to prevent an infinite loop, it is possible with such a stochastic algorithm that a photon can travel forever, often then, a cut-off is assigned where the photon is considered lost. Such terminations of photons can bias the model and in some codes extra probability models are used to determine in such cases whether photons continue. An example of the validation of Monte Carlo simulation itself for light transport can be found in [63].

More details of Monte-Carlo algorithm can be found in a variety of books and papers, for transport and other problems. An extensive review would be lengthy and somewhat unnecessary here, so we have chosen a single case from the transport literature as illustrative. A good example of a Monte-Carlo algorithm for light transport in tissue is the MLMC, or multi-layer Monte Carlo, presented in [101]. Here we see the use of photon packages instead of single photons to accelerate the process, whereby the packages are reduced in size between events depending on the coefficients at the event point. This paper also provides information on the handling of boundaries in terms of refractive index mismatch and storage of information throughout the tissue to allow for various analyses of the data to be undertaken, rather than a simple boundary model.

The important feature of such algorithms is in understanding how the path length is chosen. In the MLMC model this is done by defining the transport coefficient in terms of a probability function $\mu_t = -\frac{1}{\{P\{x\}}\frac{dP\{x\}}{dx}$, where x is the distance travelled. This can then be solved to give $P\{x\} = e^{(-\mu_t x)}$. This gives us a probability distribution function for x as $x = \frac{-ln(\epsilon)}{\mu_t}$ where ϵ is a normally distributed random number between 0 and 1.

3.4 Approximations to the RTE

In this section we will continue to consider approximating the RTE. As has already been seen, modelling the full RTE can be both expensive computationally and depending on

the model problematic in terms of what we can truly represent. As we are addressing a diffusing domain, this section will focus on what is the standard approach to this type of problem, the Diffusion Approximation. We consider this as the starting point of our work, and whilst other models exist, this is typically the model of choice in the literature.

3.4.1 Diffusion Approximation

The Diffusion Approximation (DA) is one of the most commonly known transport approximations. It can be derived in a selection of ways, and is commonly used in both neutron and optical transport problems. Ishimaru [47, 48] derives the DA from first principles using the assumptions which appear in the following derivation as limiting factors. As our overall concern is the modelling of the RTE, it is more appropriate to derive it as in [3] from the P_1 transport equation (see section 3.2.3):

$$\begin{aligned}\left(\frac{1}{c}\frac{\partial}{\partial t}+\mu_{\mathrm{a}}(\mathbf{r})\right)\Psi_{0,0}(\mathbf{r},t) &+ \frac{\partial}{\partial z}\Psi_{1,0}(\mathbf{r},t)+\frac{1}{\sqrt{2}}\frac{\partial}{\partial x}(\Psi_{1,-1}(\mathbf{r},t)-\Psi_{1,1}(\mathbf{r},t))\\ +\frac{i}{\sqrt{2}}\frac{\partial}{\partial y}(\Psi_{1,-1}(\mathbf{r},t) &- \Psi_{1,1}(\mathbf{r},t))=\mu_s(\mathbf{r})\Theta_0(\mathbf{r})\Psi_{0,0}(\mathbf{r},t)+q_{0,0}(\mathbf{r},t)\end{aligned} \tag{3.22}$$

$$\begin{aligned}\left(\frac{1}{c}\frac{\partial}{\partial t}+\mu_{\mathrm{a}}(\mathbf{r})\right)\Psi_{1,0}(\mathbf{r},t) &+ \frac{1}{3}\frac{\partial}{\partial z}\Psi_{0,0}(\mathbf{r},t)\\ &= \mu_s(\mathbf{r})\Theta_1(\mathbf{r})\Psi_{1,0}(\mathbf{r},t)+q_{1,0}(\mathbf{r},t)\end{aligned} \tag{3.23}$$

$$\begin{aligned}\left(\frac{1}{c}\frac{\partial}{\partial t}+\mu_{\mathrm{a}}(\mathbf{r})\right)\Psi_{1,-1}(\mathbf{r},t) &+ \frac{\sqrt{2}}{6}\left(\frac{\partial}{\partial x}+i\frac{\partial}{\partial y}\right)\Psi_{0,0}(\mathbf{r},t)\\ &= \mu_s(\mathbf{r})\Theta_1(\mathbf{r})\Psi_{1,-1}(\mathbf{r},t)+q_{1,-1}(\mathbf{r},t)\end{aligned} \tag{3.24}$$

$$\begin{aligned}\left(\frac{1}{c}\frac{\partial}{\partial t}+\mu_{\mathrm{a}}(\mathbf{r})\right)\Psi_{1,-1}(\mathbf{r},t) &- \frac{\sqrt{2}}{6}\left(\frac{\partial}{\partial x}-i\frac{\partial}{\partial y}\right)\Psi_{0,0}(\mathbf{r},t)\\ &= \mu_s(\mathbf{r})\Theta_1(\mathbf{r})\Psi_{1,1}(\mathbf{r},t)+q_{1,1}(\mathbf{r},t)\end{aligned} \tag{3.25}$$

In the Diffusion Approximation we will consider the quantities:

$$\text{photon density}: \quad \Phi(\mathbf{r},t)=\int_{S^{n-1}}\phi(\mathbf{r},\hat{\mathbf{s}},t)d\hat{\mathbf{s}} \tag{3.26}$$

$$\text{photon current}: \quad J(\mathbf{r},t)=\int_{S^{n-1}}\hat{\mathbf{s}}\phi(\mathbf{r},\hat{\mathbf{s}},t)d\hat{\mathbf{s}}, \tag{3.27}$$

which can be represented in the spherical harmonic forms:

$$\Phi(\mathbf{r},t) = \Psi_{0,0}(\mathbf{r},t) \tag{3.28}$$

$$J(\mathbf{r},t) = \begin{pmatrix} \frac{1}{\sqrt{2}}(\Psi_{1,-1}(\mathbf{r},t) - \Psi_{1,1}(\mathbf{r},t)) \\ \frac{1}{i\sqrt{2}}(\Psi_{1,-1}(\mathbf{r},t) + \Psi_{1,1}(\mathbf{r},t)) \\ \Psi_{1,0}(\mathbf{r},t) \end{pmatrix} \tag{3.29}$$

Making use of the following:

$$\begin{aligned} \mu_s' &= (1-\Theta_1)\mu_s \\ \kappa &= \frac{1}{3(\mu_s' + \mu_a)} \\ q &= q_0 = q_{0,0} \end{aligned}$$

Noting that $\Theta_0 = 1$ as our medium is isotropically scattering. We now get:

$$\left(\frac{1}{c}\frac{\partial}{\partial t} + \mu_a(\mathbf{r})\right)\Phi(\mathbf{r},t) + \nabla \cdot J(\mathbf{r},t) = q_0(\mathbf{r},t) \tag{3.30}$$

$$\left(\frac{1}{c}\frac{\partial}{\partial t} + \frac{1}{3\kappa(\mathbf{r})}\right)J(\mathbf{r},t) + \frac{1}{3}\nabla\Phi(\mathbf{r},t) = q_1 \tag{3.31}$$

The Diffusion Approximation follows if we assume $\frac{\partial}{\partial t}J(\mathbf{r},t) = 0$ however this is clearly untrue, but can be admitted in the case where $\mu_a \ll \mu_s$ as we have essentially a scattering medium, which makes 3.31:

$$J(\mathbf{r},t) = -\kappa(\mathbf{r})\nabla\Phi(\mathbf{r},t) \tag{3.32}$$

as $q_1 = 0$ (assumption of isotropic sources), which in turn gives rise to the Diffusion Approximation:

$$\frac{1}{c}\frac{\partial}{\partial t}\Phi(\mathbf{r},t) - \nabla \cdot \kappa(\mathbf{r})\nabla\Phi(\mathbf{r},t) + \mu_a(\mathbf{r})\Phi(\mathbf{r},t) = q_0(\mathbf{r},t) \;\; \forall \mathbf{r} \in \Omega \tag{3.33}$$

The time-independent form being:

$$-\nabla \cdot \kappa(\mathbf{r})\nabla\Phi(\mathbf{r}) + \mu_a(\mathbf{r})\Phi(\mathbf{r}) = q_0(\mathbf{r}) \;\; \forall \mathbf{r} \in \Omega \tag{3.34}$$

Boundary Conditions

The standard boundary condition for the transport equation is that only source photons travel inward at the boundary, or that the inward flux is zero, i.e.:

$$\phi(\mathbf{m}, \hat{\mathbf{s}}', t) = 0 \;\; \forall\, \hat{\mathbf{s}}'.\hat{\nu} < 0 \tag{3.35}$$

As this cannot be satisfied by the diffusion equation, we use an assumption that the inward current is zero instead, or:

$$\int_{\hat{\mathbf{s}}'.\hat{\nu}<0} (\hat{\nu} \cdot \hat{\mathbf{s}}')\phi(\mathbf{m}, \hat{\mathbf{s}}', t)d\hat{\mathbf{s}}' = 0 \tag{3.36}$$

This can be reduced to the Robin boundary condition as shown in [47, 48]:

$$\Phi(\mathbf{m}) + 2\alpha\kappa(\mathbf{m})\frac{\partial\Phi(\mathbf{m})}{\partial\hat{\nu}} = 0 \ \forall \mathbf{m} \in \partial\Omega, \tag{3.37}$$

where α is related to the refractive index mismatch.

3.5 Light Transport in Optical Tomography - a brief overview

Having discussed the general approaches to solving the RTE or approximations thereof, we should mention which models are currently in use in the case of Optical Tomography. Largely it is accepted that the Diffusion Approximation can be used in most cases [3], in fact some authors even suggest that the DA can be used in clear regions [66], we will show how this is possible, although unreliable later in this thesis.

Some authors also use Discrete Ordinates Models for this case [27, 43, 52], however as 2D results in [4] showed there may be some difficulty with this in terms of ordinate number. Further these results are in 2D and the computational expense is high, leading to concerns about the practicality of the model in 3D.

P_N models are also now being used to handle the problem of clear layers [8, 105] and other, more general, transport effects. In fact though, the P_N does experience similar difficulties to the Discrete Ordinates Model when trying to handle truly non-scattering domains in terms of order of solution and therefore computational complexity.

Monte-Carlo Simulation remains popular as although the time constraints are large the computational cost tends to be small even for more complex geometries. In fact some authors [33] have used such methods to obtain values for the scattering coefficient of the CSF, showing it to be a low and not non scattering region. These methods however are flawed by an absence of respecting the true geometry of the actual case, an absence of handling of any potential refractive index mismatch and an assumption

that all other parameters are known. Clearly in this case these results whilst potentially useful must be viewed with a certain degree of scepticism.

In summary then, we can see that most of the Transport models have been used in Optical Tomography. Most typically the Diffusion Approximation is used despite it's disadvantages in certain tissue types. However typically authors have gone to full transport models in order to solve the problem where small non-diffusive regions are incorporated. We suggest that not only is this approach limiting in terms of the domain size (due to computational cost), but that some of these models (for example the discrete ordinates method) may not provide valid results. We suggest then that of the current approaches the most reliable would be to use the Diffusion Approximation where appropriate and Monte-Carlo Methods elsewhere, thus leaving space for a new model such as the Radiosity Diffusion Model to be of good use in the field.

3.6 Summary

In this chapter we have introduced the transport equation in an easy to understand form. We have shown how modelling it has progressed through various methods in its full form, each with relative strengths and weaknesses. We concluded that in general the most appropriate full transport models were either Monte-Carlo or P_N solutions, both with large expenditure in terms of computational cost. Finally we introduced the DA as an appropriate approximation to diffusive media and explained how this is often the model of choice for Optical Tomography (OT). We also discussed that in general in OT when a non-scattering region is introduced such as the CSF, the tendency has been to opt for a full transport solution, often the DOM or Monte-Carlo. In the next chapter (chapter 4) we will go on to introduce how light is modelled in computer graphics, where we are normally dealing with non-scattering (and non-absorbing) spaces.

Chapter 4

Transport Theory - Computer Graphics

In this chapter we will introduce Kajiya's Rendering Equation (KRE), a simplification to the RTE (section 4.1). We will then go on to discuss the two classical approximations for this used in Computer Graphics (sections 4.2 and 4.3), followed by a description of some methods used to model the KRE more fully (section 4.4). Finally we will discuss the field of participating media (section 4.5) where attempts are made to model the RTE more fully. This will show that the trend in graphics has been almost the opposite of classical transport literature, moving from simpler models to more complex ones. Finally in this chapter we will discuss the visibility problem (section 4.6), which has proved to play a key role in a computationally efficient 3D model.

4.1 The Rendering Equation

In the field of computer graphics the world tends to be represented as surfaces bounding a non-participating, or vacuum region. This is because we are rendering scenes from the world as we see them day to day. It means we can, in general, approximate the RTE with a free-space form. The equation we use is referred to as Kajiya's rendering equation (KRE). This equation is given in [51] as:

$$\phi(\mathbf{m}, \hat{\mathbf{s}}) = q(\mathbf{m}, \hat{\mathbf{s}}) + \int \rho_{brdf}(\mathbf{m}, \hat{\mathbf{s}}, \hat{\mathbf{s}}')\phi_{in}(\mathbf{m}, \hat{\mathbf{s}}')d\hat{\mathbf{s}}' \tag{4.1}$$

It represents the light only on the boundaries and allows us to generate a visual representation of the domain. The bi-directional reflection distribution function (BRDF - $\rho_{brdf}(\mathbf{m}, \hat{\mathbf{s}}, \hat{\mathbf{s}}')$) represents how light is reflected at the surface. KRE approximates the

RTE by imposing the effects caused by absorption and scatter solely at the boundary of objects. Here we assume that all light is either absorbed or reflected at the surface. This means that whilst the BRDF appears initially to represent the phase function and scatter it really incorporates much more. What is important to study in graphics is how reflection functions are handled. These methods will be very useful when we come to attempt to model the full transport situation by using a similar free-space transport approach.

It is most common to adopt one of two approaches to approximating this function, Ray-Tracing, where we consider only specular reflections, or Radiosity approaches where we consider diffuse reflections. In the following two sections we will outline these two approaches and go on to some approaches to the more general case. In this thesis we will focus more on Radiosity or diffuse models as this is more relevant in our overall approach for hybrid models in diffusive media.

4.2 Ray-Tracing

Ray-Tracing is a classic reduction of KRE for handling specular reflections. It is well documented in most graphics texts (see for example [38, 94]) and well known for its images in posters of mirror balls over surfaces. In this model for light transport, we have a surface reflection function as follows:

$$\rho_{brdf}(\mathbf{m}, \hat{\mathbf{s}}, \hat{\mathbf{s}}') = \rho(\mathbf{m})\delta(\hat{\mathbf{s}} - 2(\hat{\mathbf{s}} \cdot \hat{\nu}(\mathbf{m}))\hat{\nu}(\mathbf{m})) \tag{4.2}$$

The following points are important to note and follow from this approximation to the BRDF:

- The solution to this model is view dependent and must be recalculated for each viewpoint.
- The solution is an iterative procedure involving a recursive following of the light until it reaches its destination, or becomes extinct.

In Ray-Tracing there are two classic approaches to the modelling of light, the first is Ray-Shooting and the second is Ray-Gathering. The two are essentially the inverse procedure of each other. In Ray-Shooting, we track light from the source until it reaches the image plane, in Ray-Gathering we track rays from the eye through the view-port

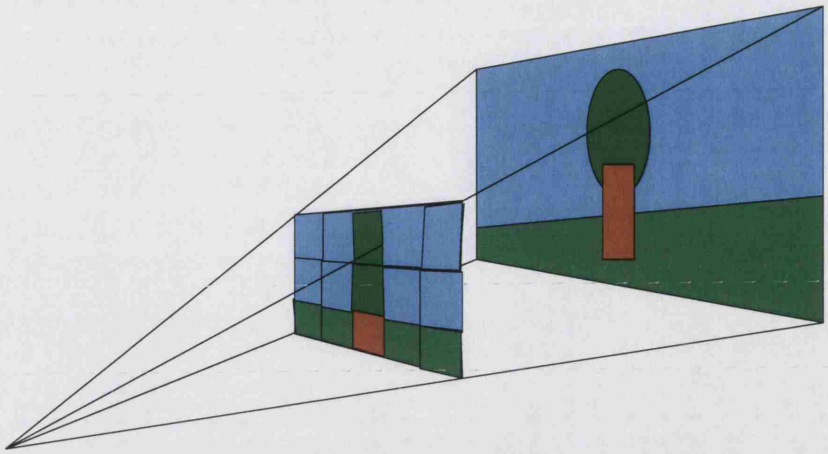


Figure 4.1: A diagram illustrating the idea of the painter's analogy Here we see the scene represented in each pixel by the colour seen at its centre.

(image plane) back into the scene. The first is clearly reminiscent of how light actually behaves, the latter can be considered as a painting analogy; the analogy is if we look though a grid of cells at our scene and then on a gridded canvas fill each cell with the colour seen through the centre of the corresponding cell in our grid we will get a pixelised image of our scene much as we get from a computer (see figure 4.1). Clearly although ray-shooting is more intuitive, ray-gathering is the more efficient approach.

We can see that ray gathering process is often expensive to get an illuminated scene, it may take many reflections to reach a light in such an approach. It is common therefore to use some more generic models to get a faster result. The usual approach here is to use a Phong shading on the light, as outlined below in section 4.4.1, we note here that such models are now not based in reality, but some approximation of it, in order to acquire an image. Arguably, however, such approaches represent more general BRDF's and therefore a better model - the point of importance is they are arbitrary, not determined physically.

4.3 Radiosity

The other most common approximation to KRE is Radiosity. This is the opposite extreme from the Ray-Tracing case, in that it assumes that light is reflected evenly from a surface. This approximation again is detailed in most graphics texts, but also has many books directly about it, for example [24, 93].

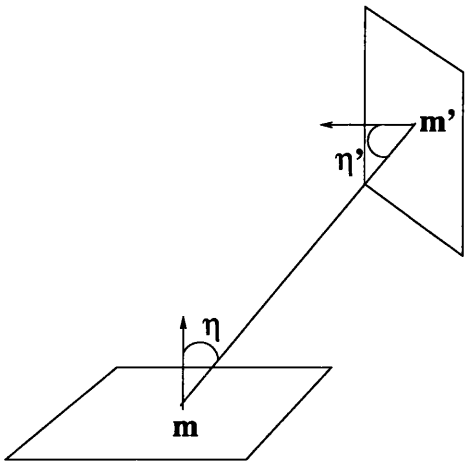


Figure 4.2: A diagram illustrating the geometry of the Radiosity problem.

4.3.1 Deriving the Radiosity Equation

Here we will derive the Radiosity Equation from the rendering equation, other similar derivations can be seen in [7, 93] etc.

Starting from equation 4.1 and using our assumption of diffuse surface reflection we can move the BRDF outside the integral. As we are assuming diffuse surfaces we must apply Lambert's Cosine Law $I(\mathbf{r}, \eta, \chi) = I(\mathbf{r})cos(\eta)$ (see [16]), which gives us:

$$\phi(\mathbf{m}, \hat{\mathbf{s}}) = q(\mathbf{m}, \hat{\mathbf{s}}) + \frac{\rho(\mathbf{m})}{\pi} \int_{\hat{\mathbf{s}} \cdot \hat{\nu} \geq 0} \phi_{in}(\mathbf{m}, \hat{\mathbf{s}}) d\hat{\mathbf{s}} \tag{4.3}$$

Now we wish to change our integration to be over the boundary, i.e. moving from angular integration over $d\hat{\mathbf{s}}'$ to $d\mathbf{m}$. To do this we must consider the geometry of our problem, see figure 4.2. Our first stage will be to consider the movement from flux to intensity we get (again using the cosine law) $\Phi(\mathbf{m}) = \int \phi(\mathbf{m}, \hat{\mathbf{s}}) d\hat{\mathbf{s}} = \int \phi(\mathbf{m}) cos(\eta) d\hat{\mathbf{s}} = \pi\phi(\mathbf{m})$. These combine to give us:

$$\Phi(\mathbf{m}) = Q(\mathbf{m}) + \rho(\mathbf{m}) \int_{\hat{\mathbf{s}} \cdot \hat{\nu} \geq 0} \phi_{in}(\mathbf{m}, \hat{\mathbf{s}}) d\hat{\mathbf{s}} \tag{4.4}$$

Now we have our equation in terms of an integral over directions, we would prefer it to be over the surface. To do this we must make the following modifications, first $\phi_{in}(\mathbf{m}, \hat{\mathbf{s}}) = \phi(\mathbf{m}', -\hat{\mathbf{s}})$.

$$\Phi(\mathbf{m}) = Q(\mathbf{m}) + \rho(\mathbf{m}) \int_{\hat{\mathbf{s}} \cdot \hat{\nu} \geq 0} \phi(\mathbf{m}', -\hat{\mathbf{s}}) d\hat{\mathbf{s}} \tag{4.5}$$

Now if we move the integral to use solid angles, ω, not directions, $\hat{\mathbf{s}} = (\eta, \chi)$, we get,

from the Jacobian of transformation:

$$\Phi(\mathbf{m}) = Q(\mathbf{m}) + \rho(\mathbf{m}) \int_{2\pi} \phi(\mathbf{m}', -\hat{\mathbf{s}}) cos(\eta) d\omega, \tag{4.6}$$

Finally if we switch integrals, we can see from geometry that $d\omega = \frac{cos(\eta')}{|\mathbf{m},\mathbf{m}'|^2} d\mathbf{m}'$, where $|\mathbf{m}, \mathbf{m}'|$ is the distance between $\mathbf{m}$ and $\mathbf{m}'$, and using $\Phi(\mathbf{m}') = \pi\phi(\mathbf{m}')$, we get:

$$\Phi(\mathbf{m}) = Q(\mathbf{m}) + \rho(\mathbf{m}) \int_{\partial\Omega} \mathrm{Vis}(\mathbf{m}, \mathbf{m}') \Phi(\mathbf{m}') \frac{cos(\eta) cos(\eta')}{\pi |\mathbf{m}, \mathbf{m}'|^2} d\mathbf{m}' \tag{4.7}$$

where we define $\mathrm{Vis}(\mathbf{m}, \mathbf{m}')$ as the Boolean visibility function:

$$\begin{aligned} \mathrm{Vis}(\mathbf{m}, \mathbf{m}') &= 1 \; if \; \mathbf{m} \; and \; \mathbf{m}' \; are \; mutually \; visible \\ &= 0 \; otherwise \end{aligned} \tag{4.8}$$

The component $\frac{cos(\eta)cos(\eta')}{\pi|\mathbf{m},\mathbf{m}'|^2}$ is often referred to as the *Form Factor*. There is a tendency to refer to integrals of this term over an area as the Form Factor and to the actual point-wise expression as the delta Form Factor.

4.3.2 Solving The Radiosity Equation

In order to solve the Radiosity equation we reduce it to a discrete form by considering $\partial\Omega$ to consist of polygons or more generally patches. Then we can reduce the Radiosity equation to a piece-wise approximation based on these polygons, as follows. From equation 4.7, if we assume the $\partial\Omega$ to consist of N polygons P_n, $n \in \{1, 2, ...N\}$, we get:

$$\Phi(\mathbf{m}) = Q(\mathbf{m}) + \rho(\mathbf{m}) \sum_{j=1}^{N} \int_{P_j} \Phi(\mathbf{m}') \frac{\cos(\eta)\cos(\eta')}{\pi |\mathbf{m}, \mathbf{m}'|^2} \mathrm{Vis}(\mathbf{m}, \mathbf{m}') d\mathbf{m}' \tag{4.9}$$

Next we assume the radiance on each patch to be a constant, which gives us:

$$\Phi(\mathbf{m}) = Q(\mathbf{m}) + \rho(\mathbf{m}) \sum_{j=1}^{N} \Phi_j \int_{P_j} \frac{\cos(\eta)\cos(\eta')}{\pi |\mathbf{m}, \mathbf{m}'|^2} \mathrm{Vis}(\mathbf{m}, \mathbf{m}') d\mathbf{m}' \tag{4.10}$$

We define the value for the constant radiosity (the exitance or source term may be defined similarly), given as the point-wise averages:

$$\Phi_i = \frac{1}{A_i} \int_{P_i} \Phi(\mathbf{m}) d\mathbf{m} \tag{4.11}$$

$$Q_i = \frac{1}{A_i} \int_{P_i} Q(\mathbf{m}) d\mathbf{m} \tag{4.12}$$

where A_i is the area of patch i. This gives us the equation:

$$\Phi_i = Q_i + \rho_i \sum_{j=1}^{N} \frac{\Phi_j}{A_i} \int_{P_i} \int_{P_j} \frac{\cos(\eta)\cos(\eta')}{\pi |\mathbf{m}, \mathbf{m}'|^2} \mathsf{Vis}(\mathbf{m}, \mathbf{m}') d\mathbf{m}' d\mathbf{m} \tag{4.13}$$

There are two ways we can approach solving this equation, directly or indirectly

Direct Approach

The direct approach is to recast the Radiosity equation as a standard invertible matrix equation:

$$(\mathbf{I} - \mathbf{F})\Phi = \mathbf{Q}, \tag{4.14}$$

where

$$F_{ij} = \frac{\rho_j}{A_i} \int_{P_i} \int_{P_j} \frac{\cos(\eta)\cos(\eta')}{\pi |\mathbf{m}, \mathbf{m}'|^2} \mathsf{Vis}(\mathbf{m}, \mathbf{m}') d\mathbf{m}' d\mathbf{m}$$

Using existing methods this model can now be solved to completely describe the illumination in a scene at the cost of solving a large dense matrix equation.

Indirect Approach

Indirect methods work by iteratively propagating light around the scene. The radiance values Φ_i and source values Q_i are stored at every point. We start with the source values, and using the form factors transfer the energy to all visible polygons. This gives us a temporary Φ_i value at all these points and reduces the source Q_i value to zero. It also turns all these points, via their reflectance values, into source terms giving each patch an effective Q_i. We can propagate the energy through the scene by iterating this procedure, a process which can be continued until the total Q_i reach zero or fall below some predetermined threshold.

This approach is used for Radiosity in graphics for a number of reasons:

- You don't need to store and solve a large matrix equation which can be computationally expensive.
- You can stop at some arbitrary point where the scene looks good enough, if you are unconcerned about exact values.
- You can display the scene after each iteration, or after some number of iterations as a gradual refinement process.

This method is similar to an iterative matrix solve, it has the advantage however that it does not store the whole matrix. This means a reduced memory storage, and in some cases where the scene contains very dark areas some matrix values will never be calculated. Also as will be seen from some of the approaches to solving the problem, it also allows for easy re-partitioning of the scene.

Our Approach

In our work we have focused on direct formulations of the problem. This is because we are already considering a system where we intend to solve some approximation of the RTE directly. In this case some of the methods used for calculation of Form Factors become inappropriate, for example the Hemi-Cube method proposed in [23], or progressive refinement techniques such as suggested in [22]. However we shall discuss these in the following section as they have influenced our work in terms of examining the visibility problem and/or mesh generation, two of the key aspects of providing a stable model.

4.3.3 Solving the Form Factor

The Radiosity equation has been studied for many years in Computer Graphics and a host of methods for solving the Form Factors exist. Most of these approaches involve speeding up the solution, or somehow optimising the number of calculations by controlling the subdivision of $\partial\Omega$ such that it is optimal for the light source. Our problem then becomes twofold. First we are using direct models, so as already mentioned, some of these techniques do not apply. Secondly in terms of optimising our space we have two problems, first we are using a mesh based on a volume (a space in $\mathbb{R}^3$) as opposed to a surface (a manifold in $\mathbb{R}^2$). As mentioned in section 2.2, this is a much harder problem and we are more constrained by generating a good mesh, giving us less control over how we distribute the size of its elements. The second problem is that if we consider the application of OT we will require more than one light source, meaning we will need multiple optimisations for our problem, either implying the use of different meshes (introducing the problem of how to maintain accuracy and shape), or one more complicated and hence larger (i.e. numerically and computationally more difficult to solve) mesh.

In the following sections we will outline some of the more common approaches

which have been developed for solving the Form-Factor, and/or optimising the process of modelling the domain.

The Hemi-Cube

The Hemi-Cube is an idea presented in [23], based on a remapping of the domain to a Hemi-Cube. The idea originates from an understanding of the meaning of the Form Factor and using this to create an equivalent model of the world via projection. This approach has two key advantages, first it makes the visibility determination and Form Factor calculation a single process and second, as we use a uniform mapping the numerics of the Form Factors become trivial.

The Hemi-Cube works as follows:

- find patch with largest unshot radiosity (this is either source light or light which has been transmitted to, but not yet reflected from, a patch)
- project the visible world onto a Hemi-Cube covering this patch
- using a pixel grid on this Hemi-Cube associate each pixel to the patch visible to that pixel
- given the projection relationship described below, add each pixels' Form Factor to its associated patch
- transmit unshot radiosity
- step to next highest unshot radiosity patch and continue until threshold

The Hemi-Cube simplifies the calculation of the Form Factor as we simply calculate a Form Factor locally. We can understand this if we consider figure 4.3. Here we see an illustration showing how the energy exchange from any surface (A) to another (B) is equivalent to the energy exchange between the first (A) and the projection of the second (B) onto the Hemi-Cube (C).

The primary problem with the Hemi-Cube model is that of aliasing. If our sampling is insufficient the Hemi-Cube algorithm can break down, this is illustrated clearly in figure 4.4 where we see an object falling between cells of the Hemi-Cube. Here as each pixel or division only sees the object at its centre, object B is made not visible at

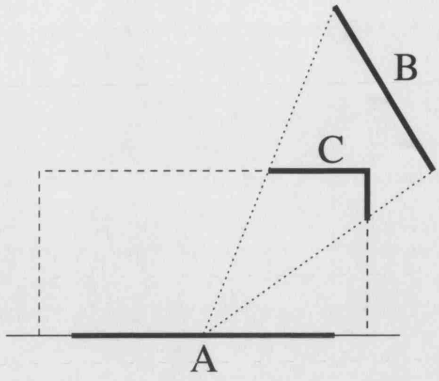


Figure 4.3: A diagram illustrating how the Hemi-Cube simplifies the Form Factor calculation.

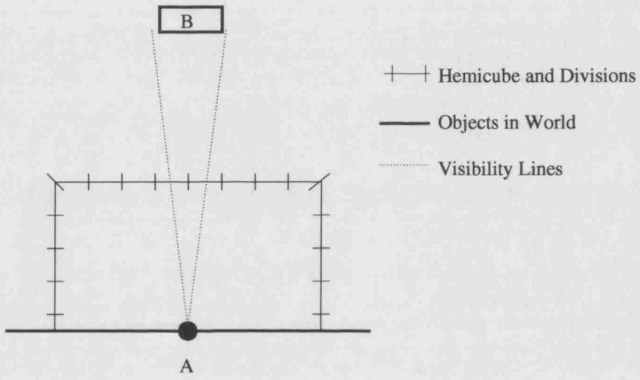


Figure 4.4: A diagram illustrating how aliasing effects the Hemi-Cube approach.

point A, which is clearly an error. This combined with the patch to patch nature of the hemi-cube provide problems in our domain, we will see this more clearly later when we consider how to effect a reasonable model in the 3D RDM. We note, however, that the Hemi-cube approach has provided inspiration for one of our models for visibility calculation, which whilst not entirely effective in a generic domain is suitable and rapid for a constrained set of domains.

Progressive Refinement

The progressive refinement technique is another common approach to simplifying the calculation process, it is outlined in [22]. Here the idea is to create a coarse representation of the world and refine it as necessary, thus minimising the unnecessary work done.

The principle of the refinement process is as follows, starting from a solution on a

coarse mesh we progress following the algorithm:

- Check all radiosity gradients (the difference of neighbouring patch radiosity values) between patches.
- for all patches with large gradients at the edge subdivide
- recalculate radiosity distribution over new subdivisions

This process is repeated to a certain number of subdivisions, specified to prevent a continuous loop caused by a hard shadow boundary. A second reason for stopping is that we may have reached a reasonable image. As we are only concerned with the appearance of our scene in graphics, there is no point in continuing this process in regions where the induced change is not visually significant.

Whilst this model does have clear computational advantages it is unclear where it will be useful in our model. As we have a very diffuse light situation, we will not be dealing with hard boundaries, which is good. However as we may be using multiple light sources we would need to repeat this process for each source, so while we may reduce our costs in one regard (memory), we do stand to have a significant time overhead added to our solution process. Also we are disadvantaged, as already mentioned, by the re-meshing process, whilst in the 2D manifolds of computer graphics this is not a real difficulty we are dealing with 3D volume meshes, where re-meshing can be extremely difficult and time-consuming.

Analytic Models

It has been seen in the literature that an analytic approach can be taken to solving the Form Factor [84, 50]. In the early work [84] this was done using purely planar patches, but was developed in [50] to handle spherical patches - patches which form part of a spherical surface.

The obvious limitations of such analytic models are threefold, as follows:

- The nature of the patches is highly restrictive to the geometries which can be expressed - currently only planar polygons [84] or patches which conform to a sphere [50] can be handled.

- The current patch-wise nature of the model is such that it is appropriate only really to point collocation models - which we will see in chapter 7 are not the best for our approach to handling the domain.

- A requirement of the model is that patches are completely mutually co-visible.

Having spent some time considering such models we would propose that whilst the first two of these problems are likely soluble, the third is more restrictive. In fact we cannot see a way around this, whilst in some geometries/applications it may be possible to ignore slight discrepancies in the visibility to be acceptable, we do not see this approach as a generic one which would allow us to make any statements about the problem. It is suggested that such an investigation may prove suitable as a future doctoral research project in it's own right.

Higher-Order FEM approaches

If we consider the classical form of the solution of Form Factors in Radiosity we can see that it is essentially a zero order FEM approach, that is we consider functions to be constant over patches (elements) of the problem domain. It has been suggested [99, 107] that a higher order approach to solving the Form Factors is a more suitable methodology.

In [107] we see a Galerkin higher order model, this is expanded on in [99] to include a comparison of point collocation and Galerkin approaches to higher order representation. The results presented in [99] suggest that a point collocation approach provides a better solution than a Galerkin approach. On closer examination however we note the baseline for the comparison is a Hemi-Cube algorithm, which by its nature will provide a more similar error to the point collocation method. In this work we will investigate both models, and draw conclusions about their effectiveness in our problem.

Shadow-Masking

Another interesting development when considering higher order approximations is introduced in [107], this is the idea of shadow masking. Shadow masking refers to the idea of using the higher order basis functions to represent visibility partially over a patch rather than as a single valued function across it. In [107] the author suggests creating a partial visibility fraction to multiply with the Form Factor of a patch by inte-

grating the visibility as a separate function from the Form Factor, this can be expressed as:

$$F_{ij}(\mathbf{m}) = \int_{P_i} \frac{cos(\eta)cos(\eta')}{\pi|\mathbf{m},\mathbf{m}'|} d\mathbf{m}' \int_{P_j} \mathrm{Vis}(\mathbf{m},\mathbf{m}')d\mathbf{m}' \tag{4.15}$$

However this misses some of the elegance available to the FEM approach as it ignores the relative scale of the Form Factors at each point. That is, by considering the visibility to be part of the Form Factor integral formulation, instead of a separate one, we relate the proportional visibility of the patch with the magnitude of the Form Factors across it. Consider if we have a patch which is visible at only one end but the Form Factor here is small and large at the non-visible end, separating the visibility integral from the Form Factor one would give weight to the large Form Factor region whereas a combined integral would remove this error. Later we will discuss a single integration approach which should improve our calculations, reduce the computational overhead and potentially reduce the scale of meshing required to solve our problem. It can be understood simply by reverting equation 4.15 to the original formulation for radiosity:

$$F_{ij}(\mathbf{m}) = \int_{P_j} \frac{cos(\eta)cos(\eta')}{\pi|\mathbf{m},\mathbf{m}'|} \mathrm{Vis}(\mathbf{m},\mathbf{m}')d\mathbf{m}' \tag{4.16}$$

4.4 Generic Reflection distributions

Obviously the solutions so far discussed in graphics only apply to special cases of the BRDF, what however if we wish to use a general form for the BRDF. In this section we will discuss some of the approaches used to handle more general forms of the BRDF. It should be noted that in graphics such methods tend to have arbitrary approaches to handling the BRDF, although it is becoming more common to extract the BRDF from objects via computer vision techniques. Here we will mention a couple of these generalised BRDF techniques, although they are not essential to our current work, they are more relevant to further potential developments of the RDM to with refractive index mismatches. We include them here for completeness and to illustrate that such concepts should be easily incorporated in future to our model.

4.4.1 Phong Shading and ambient light

The idea behind Phong shading is to allow for more generic reflectance distributions, by admitting non-perfect specular reflections to the model. An example of a Phong

model for light at a point in a ray-tracing model is as follows:

$$\Phi_t = \Phi_a + \Phi_s + \Phi_{s'} \tag{4.17}$$

This implies that the total radiance, Φ_t, is a sum of an ambient term, Φ_a, a specular term, Φ_s, and an off-specular term, Φ'_s. The specular term will come from true specular effects from objects, while the off-specular term will come from any lights which are just off the perfect specular ray. That is to say, in normal ray-tracing lights will only be seen of they are exactly in the specular reflection path, this approach allows lights to be incorporated if they are near the true path giving a more general, though artificial, lighting model. The method used to generate the non-perfect specular effects is to use a weighting function. One method to calculate the off-specular reflections from the lights is as follows:

$$\Phi_{s'} = Q\rho_s(\mathbf{m})cos(\upsilon) \tag{4.18}$$

where υ is the angle between the source and the true specular reflection angle, Q is the off-specular light and $\rho_s(\mathbf{m})$ is the specular reflection co-efficient on the surface.

The ambient light term is used to give colour to the scene in dark areas in such a way as to reduce the need for deep ray-casting. The ambient term is completely unphysical in meaning and is just an approach used to speed up realistic image synthesis.

4.4.2 Hybrid Models

Another possible approach is to combine a Radiosity and Ray-Tracing Model, see for example [62]. Here they provide a solution which combines the Radiosity and Ray-Tracing components into a single model. They do this by having two reflection co-efficients, specular and diffusive, then they simply implement a ray-tracing and a radiosity model to calculate scene illumination, noting that there will be interactions between the two. This is often done by using an iterative shooting algorithm.

4.5 Beyond the Rendering Equation

In graphics more recently it has become common to move from an implementation of KRE to include other effects. This has been motivated by two factors, gaseous effects such as smoke, providing scattering in the model, or caustic (sub-surface) effects such as marble which generate effects which cannot be represented by a surface reflectance

model. Reviews of these can be found in [68, 76]. Here we will outline two of the types of approach which have been taken to this problem, stochastic and numerical.

4.5.1 Stochastic Models

In terms of stochastic approaches two stand out as examples in the Graphics literature, standard Monte-Carlo type simulation (see for example [95]) and Photon-Mapping (see for example [49]).

Monte-Carlo simulation

Stam's work, [96, 97], is quite extensive and covers a stochastic model with much scope as well as some numerical models. Here we are considering his statistical approach as it is of most interest. We note for interest that he does (as has been done in Optical Tomography) also compare the Diffusion Approximation to a Discrete Ordinates Model.

Stam's approach is to use a stochastic model to update the scene in two ways, he is modelling both the movement of particles (smoke for example) and the lighting of the scene. As our interest is in a lighting model we will only examine his model in terms of the illumination question rather than its motion aspects. Stam's model uses a ray-tracing scheme as outlined in section 4.2 thereby computing the probable lighting in a scene, it modifies each ray as it passes through a participating medium by its transparency t and by its average diffuse light (the scattering source term). This latter diffuse light term is calculated using an algorithm similar to the zonal method (outlined in section 4.5.2 below). Fundamentally Stam's approach is a scheme whereby we take a basic ballistic effect and add in extra information about the probable in-scattered light.

Photon Mapping

Photon Mapping [49] was a technique developed in graphics to handle generic reflectance models. Its starting assumption is to accept that whilst Finite Element techniques are more accurate, they are in terms of producing real-time image synthesis, too computationally expensive. The idea here is to combine a two stage process of tracking photons out of light sources and reading them back to the view point. In this way we develop a map of light within a domain. It allows us to store general diffusive information, but to generate the specular components as we go along with a changing viewpoint.

The model then uses a stochastic process to build up the light transport in a scene,

using techniques similar to a Monte-Carlo simulation. It stores diffusive exchange in the scene in a photon map, a spatial representation of the scene independent of the geometry. In order to calculate the actual illumination it then adds in the specular component using classical ray approximation. This is separated as whilst feeding information from both ends of a photon's path is statistically possible with diffuse reflectors, the odds of a photon bundle leaving from source and detector arriving at the same point with the correct directional component is increasing small as the reflectance becomes more specular, in fact it becomes effectively zero once a perfect specular BRDF is reached.

The primary advantage of this model is to incorporate the strengths of a static model where possible, but to include a sweeping Ray-Trace for those components where it is necessary. The disadvantage of such a model in our case is we have been primarily interested in a diffusive scene, which ignores really the greatest strength of the photon map. Also as the author himself admits, we are seeking to move away from a necessarily accurate model, though photon mapping does produce reasonably accurate results, it is not the goal of the model, and therefore we are looking at a model whose design paradigm is almost contrary to that which we seek. We suggest that photon-mapping may be a technique to examine as our light transport models become more complex, including more generic domains, but at this stage its' advantages are in areas which will not be exploited by the scope of domains we are considering.

4.5.2 Numerical Approaches

One of the earliest approaches to the problem of scattering inclusions in graphics was a numerical approach called the zonal method. It was originally developed by Rushmeier et al. in [78], and is extended to a hierarchical model by Sillion in [91]. It can be considered as a method for extending the Radiosity model to include scattering regions by solving a full transport solution assuming isotropic phase functions. Here we present a brief outline of how the model and its associated equations are formulated.

We define a domain bounded by diffuse surfaces, within this world we have volumes of scattering particles. As such we have the equation:

$$\Phi(\mathbf{r}) = Q(\mathbf{r}) + \int_{\Omega} f(\mathbf{r}, \mathbf{r}')\Phi(\mathbf{r}')d\mathbf{r}' \tag{4.19}$$

where we must define some function $f(\mathbf{r}, \mathbf{r}')$. This is done in a discrete formulation

using a zero order basis over patches and volumes, giving us:

$$\Phi_i = Q_i + \sum_{j=i}^{N} f_{ij}\Phi_j \tag{4.20}$$

where f_{ij} describes the proportion of light at j contributing to i, and Φ_i and Q_i are as previously defined .

The zonal method assumes isotropic scatter in the medium, so from this we can calculate the coupling factor f_{ij}. As usual from the Radiosity approach we know we must include the Boolean, $\mathrm{Vis}(i,j)$, and basic transport tells us we must include a term for the loss over the separation, $e^{-\mu_t|i,j|}$. Next we must calculate the geometric contributions of each patch/volume to each other patch/volume. There are four cases we must consider, they are patch to patch, patch to volume, volume to patch and volume to volume. We have already established the patch to patch term in the Radiosity Model, and the others follow by analogy to give us:

$$\textit{Patch to Patch}: \quad f_{ij} = \frac{1}{A_i}\int_{\tau_i}\int_{\tau_j} \rho(\mathbf{m})\mathrm{Vis}(\mathbf{m},\mathbf{m}')e^{-\int_{\mathbf{m}}^{\mathbf{m}'}\mu_t(\mathbf{x})d\mathbf{x}} \times \frac{cos(\eta)cos(\eta')}{\pi|\mathbf{m},\mathbf{m}'|^2}d\mathbf{m}'d\mathbf{m} \tag{4.21}$$

$$\textit{Patch to Volume}: \quad f_{ij} = \frac{1}{A_i}\int_{\tau_i}\int_{\tau_j} \rho(\mathbf{m})\mathrm{Vis}(\mathbf{m},\mathbf{r})e^{-\int_{\mathbf{m}}^{\mathbf{r}}\mu_t(\mathbf{x})d\mathbf{x}} \times \frac{\mu_s(\mathbf{r})cos(\eta)}{\pi|\mathbf{m},\mathbf{r}|^2}d\mathbf{m}d\mathbf{r} \tag{4.22}$$

$$\textit{Volume to Patch}: \quad f_{ij} = \frac{1}{A_i}\int_{\tau_i}\int_{\tau_j} \mathrm{Vis}(\mathbf{m}',\mathbf{r})e^{-\int_{\mathbf{m}}^{\mathbf{r}}\mu_t(\mathbf{x})d\mathbf{x}} \times \frac{cos(\eta')}{\pi|\mathbf{m}',\mathbf{r}|^2}d\mathbf{r}d\mathbf{m}' \tag{4.23}$$

$$\textit{Volume to Volume}: \quad f_{ij} = \frac{1}{A_i}\int_{\tau_i}\int_{\tau_j} \mathrm{Vis}(\mathbf{r},\mathbf{r}')e^{-\int_{\mathbf{r}}^{\mathbf{r}'}\mu_t(\mathbf{x})d\mathbf{x}} \times \frac{\mu_s(\mathbf{r}')}{\pi|\mathbf{r},\mathbf{r}'|^2}d\mathbf{r}d\mathbf{r}' \tag{4.24}$$

where A_i is the area for a patch and the volume for a volume element. This set of equations is discretised as for the Radiosity Equation and then solved in a similar fashion. We note that should the scattering in the volumes be reduced to zero, it would imply that the intensity on the surfaces would be the same as for the Radiosity equation, but that we could also evaluate the intensity at a point in the volume from the components given by equation 4.22.

4.6 Visibility Issues

One aspect of the Radiosity problem is the visibility question. In and of itself the meaning of visibility is simple and is an easy problem to solve. The main issue becomes how to do this for a general case in an efficient manner. Over the years there have been many approaches to solving this problem, for example [28, 32, 31, 34, 40, 67, 92, 104], for which a classification system can be found in [61], the question is finding an appropriate model to the given domain and solution method.

In this section we will first introduce the basic principle and introduce the simple approach to solving it. We will then go on to discuss a variety of classical approaches to make solving it more efficient and how they would apply in our case. It is important to note that in a case such as the Hemi-Cube solution for Form Factors (see section 4.3.3) the visibility model is implicit in the solution and no such calculation is needed.

4.6.1 The Visibility Problem and the trivial solution

The visibility problem can be expressed as follows:

$$\begin{aligned} \mathrm{Vis}(\mathbf{i},\mathbf{j}) &= 1 \text{ iff } \mathbf{i}+\mathrm{x}(\mathbf{j}-\mathbf{i}) \notin \partial\Omega \,\forall\, ((\mathrm{x}>0)\ \&\&\ (\mathrm{x}<|\mathbf{j}-\mathbf{i}|)) \\ &\quad \&\&\ \mathbf{i}+x(\mathbf{j}-\mathbf{i}) \in \Omega \,\forall\, ((x>0)\ \&\&\ (x<|\mathbf{j}-\mathbf{i}|)) \\ &= 0 \text{ otherwise,} \end{aligned} \tag{4.25}$$

if $\partial\Omega$ is a C^1 continuous surface this can be simplified to:

$$\begin{aligned} \mathrm{Vis}(\mathbf{i},\mathbf{j}) &= 1 \text{ iff } \left(\frac{\mathbf{i}\cdot(\mathbf{j}-\mathbf{i})}{|\mathbf{i}||\mathbf{j}-\mathbf{i}|}>0\right)\ \&\&\ \left(\frac{\mathbf{i}\cdot(\mathbf{j}-\mathbf{i})}{|\mathbf{i}||\mathbf{i}-\mathbf{j}|}>0\right) \\ &\quad \&\&\ \mathbf{i}+x(\mathbf{j}-\mathbf{i}) \notin \partial\Omega \,\forall\, ((x>0)\ \&\&\ (x<|\mathbf{j}-\mathbf{i}|)) \\ &= 0 \text{ otherwise.} \end{aligned} \tag{4.26}$$

We note that if $\partial\Omega$ is described as a simple set of analytic surfaces such as spheres, we can easily write down an equation which solves the visibility model.

The easiest way to calculate visibility in a polygonal domain extends from the idea above. If we have $\partial\Omega$ which consists of a set of polygons,$\{\tau_i; i = 1, 2, ...T\}$ all we must do is check the intersection of the line joining our two points with each polygon τ_i in $\partial\Omega$ and see if it lies between the two points. This process is inefficient as it ignores any knowledge of the geometry of our domain that we may have. Most of the following systems will use some kind of pre-determined knowledge about the scene in order to reduce the computational overhead of the problem.

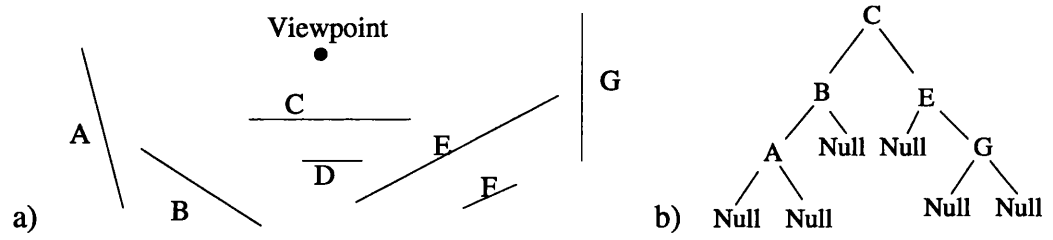


Figure 4.5: A figure illustrating the workings of the BSP-Tree structure, a) showing a 2D scene of labelled polygons and a viewpoint and b) the structure of a BSP-Tree generated from it.

4.6.2 Binary Space Partition Trees

Typically visibility models seek to somehow reduce the computational burden using a structured division of our domain. The Binary Space Partition Tree or BSP-Tree is one of the best known and most studied approaches. It relies on dividing the scene into a logical tree based on building up a set of visible edges.

The idea is to start from the view point, $\mathbf{i}$, and build up a map of the visible space, i.e. find all $\mathbf{j}$ where $\text{Vis}(\mathbf{i}, \mathbf{j}) = 1$. This is done by first sorting the objects by proximity to the viewpoint, then to create a list of the edges of visible space by testing whether the next object is completely occluded or partially occluded by what is already present. Figure 4.5a illustrates a 2D scene with a set of objects within it, figure 4.5b illustrates the tree structure for this scene. The secondary leaves are to allow us to fill in holes in a scene where object x is to the left of object y but there is a gap between them and then we can fill this with object z which is to the right of x and further back in the scene, but is visible in the gap.

The BSP-Tree is a rapid way to grow a visibility structure from a given viewpoint, we can easily see that there are relatively few tests required to place a new object in the scene as the tree grows to determine visibility. The problem with such models is that they are viewpoint dependent, some steps have been made towards allowing the viewpoint to gradually change, and given a well structured scene the new tree should be faster to generate as we have already a structure to our model. Given that we are trying to develop the visibility to the entire scene from all viewpoints, it was noted that this was not how this algorithm was optimised. Therefore other approaches were considered that were designed towards our type of problem.

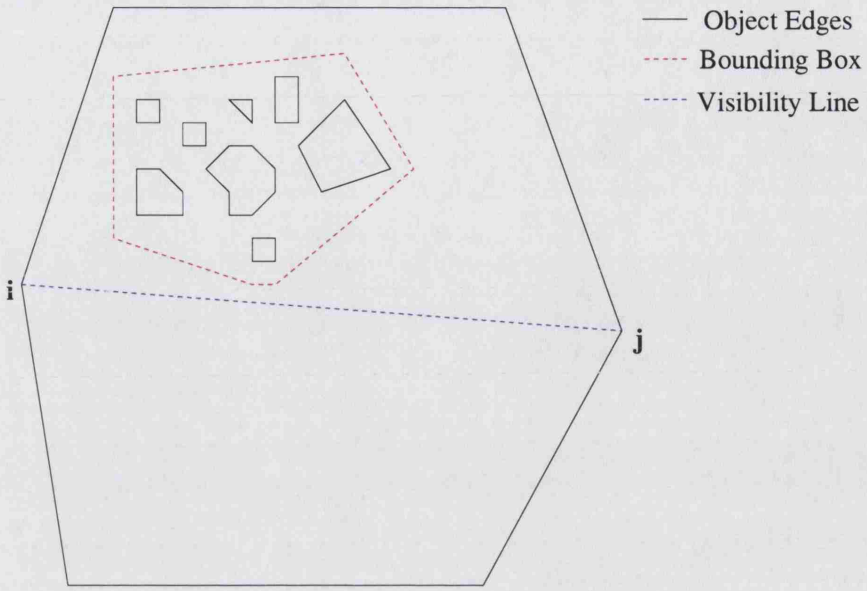


Figure 4.6: A figure illustrating a set of objects being eliminated from a visibility search between points **i** and **j** by use of a bounding box in an object hierarchy.

4.6.3 Spatial Sub-Division Schemes

Spatial subdivision schemes come in two main approaches. The most common approach is from ray-tracing models, and involves building a scene hierarchy, the second is to use some kind of structured model of the scene to reduce the number of objects tested. Here we will outline both approaches in order to explain our choice of the latter type for our second visibility model.

Object Based Hierarchical Sub-division

Object based hierarchies are designed for domains containing multiple discrete objects. The idea is if you have a space containing multiple objects, you can effectively reduce the visibility computation by introducing a hierarchy of spatial sets. The method is designed to handle the idea that if we have a cluster of small objects in one region of space, then if we enclose them in a bounding box which we test for intersection with the line between two points first we can potentially eliminate a large number of calculations from our visibility computation (see figure 4.6 where a large set of objects can be eliminated easily from our search for the visibility of points **i** and **j**). If we construct a hierarchical scene graph in this manner, we can hopefully optimise our visibility search

The drawback of this model is we are considering multiple object groupings. in our scene we tend to have single objects constructed from multiple polygons. At first this seems an obvious extension to this idea, until we consider the low relative curvature of our surfaces. With a low curvature, in order to include any large number of polygons, our bounding volumes (if kept trivial in order to maintain ease of intersections tend to become large relative to their contained objects - i.e. intersections may occur which are due to an extension of the bounding box rather than a true reflection of geometrical properties.

Cartesian Spatial Sub-division

In this scheme we are essentially dealing with an approach which discretises our search space. An example of the type of model we will describe here can be found in [2], although here it was used for ray-tracing. The approach here is to use a regular grid division in space instead of a structural division. There is a second reason for using such an approach, not only can we discretise the location of objects in the scene, but we may also use a simple traversal algorithm, such as a Bresenham algorithm [18] to traverse between two points to check for visibility. Provided we are intelligent in our storage of association between our spatial subdivision and our objects within it, this gives us a fast access to a restricted set of objects which will determine if the two points are visible. In our model such an approach is well suited, and as we shall see later provides a suitable model for our domain.

The Bresenham algorithm constructs a trace ray through a voxel space. Usually it uses six connectivity for speed, but in our approach to ensure a complete set of voxels for the connecting line we use a 26 connectivity. This then creates a traces through the domain and we can examine only polygons associated with our selected voxel set. In order to accelerate the search we use a sparse matrix structure, so rather than checking off whether each side is associated with our voxel we simply scan an associated list of sides.

Another advantage of such a model, is where we consider minor geometric updates to our domain. If we wish to slightly alter our domain, we can easily access which of our spatial units have a changed geometry, and therefore which pairs of points will require a visibility update. This will not be of importance in this work, but if it is to be

extended to modelling the inverse problem more effectively, this may be considered an extremely useful property.

4.7 Summary

In this chapter we have outlined the development of light transport models in computer graphics. We have examined how they have developed from simple free-space models to more complex fuller transport solutions. We have examined in more detail aspects of the Radiosity model a free-space model which is based on having diffusive boundaries. We have seen that in graphics the opposite of the development in the transport literature, from simplifications to full solution rather than the other way around. In the next part of this work we shall examine how taking idea's from both these fields, we can develop a hybrid to handle our chosen problem - diffusing domains containing non-scattering spaces. In doing this we will use existing idea's from both fields and some novel idea's to create a working model for the 3D problem.

Part II

Deriving the Radiosity Diffusion Model

Chapter 5

The Radiosity Diffusion Model - Theory

In this chapter we will introduce the theoretical aspects of the Radiosity Diffusion Model. We will show its derivation first (section 5.1), followed by some analytic models we have used for validation of our numerical approaches (section 5.2). We will continue by giving the theory behind our two geometric models to reduce computation costs, localised p-refinement (section 5.3) and the simplified meshless method (section 5.4) and finally we will discuss some similar approaches to this problem from the current literature (section 5.5). In chapter 6, we will go on to discuss the numerical approaches we have developed to extend this model into three dimensions, some of which have been successful, whilst others have been less so, for technical or purely numerical reasons.

5.1 Deriving the RDM

In this section we will derive the Radiosity Diffusion Model, we will start with a quick revisitation of the Diffusion Approximation. Here we will show how it can be modelled using a FEM approach (as in [6, 86]). We will follow this by an explanation of how we will handle the non-scattering spaces, and finally we will show how we derive the non-local boundary condition, we do this slightly differently from its original derivation in [4], as it stems from the Radiosity approach in graphics, however we arrive at the same boundary condition. The advantage of this approach is it explicitly links the transport approach to graphics and gives us confidence to freely borrow techniques from both domains to solve the problem at hand. Finally we will derive the complete

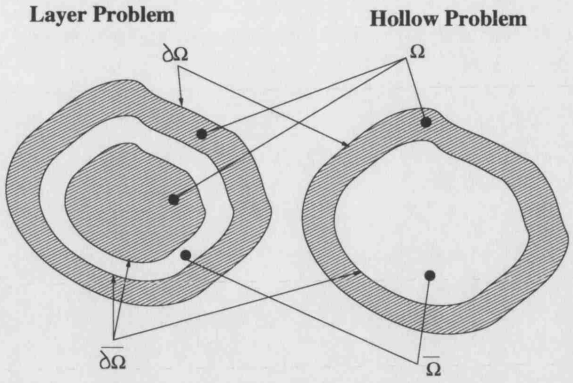


Figure 5.1: A diagram illustrating the two types of void inclusion we will consider in this work, a thin layer and a hollow region.

RDM equations for a FEM approach.

5.1.1 Defining our domain

In deriving our model we must first examine the domain under consideration. To do this we will need to make some more specific notation as follows:

- Ω will now represent a domain where the diffusive regime holds.
- $\partial\Omega$ will represent the external boundary of this domain.
- $\overline{\Omega}$ will represent the non-scattering inclusion(s) within Ω.
- $\partial\overline{\Omega}$ will represent the boundary(ies) of said inclusion(s).

We know from chapter 3 that regions such as Ω can be modelled using the Diffusion Approximation. We have also seen that regions such as $\overline{\Omega}$ can be modelled using some form of Kajiya's Rendering Equation. In the following sections we will lay out how we combine these to form the RDM a novel model for handling our problem. We illustrate the types of domain of interest in figure 5.1, clearly showing the two types of problem which may arise, clear layers and hollow regions. These relate in Optical Tomography to the sub-arachnoid space and the ventricles respectively.

5.1.2 Diffusion Approximation Revisited - modelling Ω

We start our derivation from the DA as defined previously (equation 3.33), which we re-state here for convenience:

$$-\nabla \cdot \kappa(\mathbf{r})\nabla\Phi(\mathbf{r}) + \mu_a(\mathbf{r})\Phi(\mathbf{r}) = q_0(\mathbf{r}) \;\; \forall \mathbf{r} \in \Omega, \tag{5.1}$$

With a Robin boundary condition given by:

$$\Phi(\mathbf{m}) + 2\alpha\kappa(\mathbf{m})\frac{\partial\Phi(\mathbf{m})}{\partial\hat{\nu}} = 0 \ \forall \mathbf{m} \in \partial\Omega. \tag{5.2}$$

5.1.3 FEM formulation of the DA

Starting from equation 5.1, we perform the usual PDE approach of integration with a test function to obtain the weak, integral form:

$$\int_\Omega -\Psi(\mathbf{r})\nabla\cdot\kappa(\mathbf{r})\nabla\Phi(\mathbf{r})d\mathbf{r} + \int_\Omega \Psi(\mathbf{r})\mu_a(\mathbf{r})\Phi(\mathbf{r})d\mathbf{r} = \int_\Omega \Psi(\mathbf{r})q_0(\mathbf{r})d\mathbf{r} \tag{5.3}$$

Integrating by parts, we get:

$$\begin{aligned}\int_\Omega \kappa(\mathbf{r})\nabla\Psi(\mathbf{r})\cdot\nabla\Phi(\mathbf{r})d\mathbf{r} \ &+ \ \int_\Omega \Psi(\mathbf{r})\mu_a(\mathbf{r})\Phi(\mathbf{r})d\mathbf{r} \\ &= \int_\Omega \Psi(\mathbf{r})q_0(\mathbf{r})d\mathbf{r} + \int_{\partial\Omega} \Psi(\mathbf{m})\kappa(\mathbf{m})\frac{d}{d\hat{\nu}}\Phi(\mathbf{m})d\mathbf{m}.\end{aligned} \tag{5.4}$$

The surface flux is obtained directly from equation 5.2:

$$\frac{d}{d\hat{\nu}}\Phi(\mathbf{m}) = -\frac{1}{2\alpha\kappa(\mathbf{m})}\Phi(\mathbf{m}) \tag{5.5}$$

Leading to:

$$\begin{aligned}\int_\Omega \kappa(\mathbf{r})\nabla\Psi(\mathbf{r})\nabla\Phi(\mathbf{r})d\mathbf{r} + \int_\Omega \Psi(\mathbf{r})\mu_a(\mathbf{r})\Phi(\mathbf{r})d\mathbf{r} \ &+ \ \int_{\partial\Omega}\frac{1}{2\alpha}\Psi(\mathbf{m})\Phi(\mathbf{m})d\mathbf{m} \\ &= \int_\Omega \Psi(\mathbf{r})q_0(\mathbf{r})d\mathbf{r}.\end{aligned} \tag{5.6}$$

which can be discretised and solved in the standard way using the Finite Element Method:

$$(\mathbf{K}(\kappa) + \mathbf{C}(\mu_a) + \frac{1}{2\alpha}\mathbf{A})\mathbf{\Phi} = \mathbf{Q} \tag{5.7}$$

with system matrix and vector entries:

$$K_{ij} = \int_\Omega \kappa(\mathbf{r})\nabla u_i(\mathbf{r})\cdot\nabla u_j(\mathbf{r})d\mathbf{r} \tag{5.8}$$

$$C_{ij} = \int_\Omega \mu_a(\mathbf{r})u_i(\mathbf{r})u_j(\mathbf{r})d\mathbf{r} \tag{5.9}$$

$$A_{ij} = \int_{\partial\Omega} u_i(\mathbf{m})u_j(\mathbf{m})d\mathbf{m}, \tag{5.10}$$

$$\Phi_i = \int_{\partial\Omega} \phi_i u_i(\mathbf{r})d\mathbf{r}, \tag{5.11}$$

$$Q_i = \int_{\partial\Omega} q_j u_i(\mathbf{r})d\mathbf{r}, \tag{5.12}$$

where $u_i(\mathbf{r})$ and $u_j(\mathbf{r})$ are the Finite Element basis functions associated with nodes i and j.

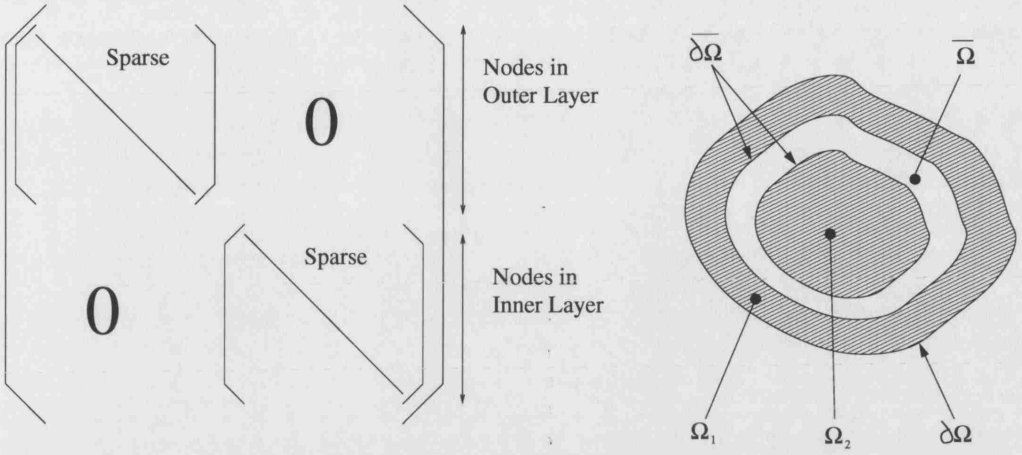


Figure 5.2: A diagram illustrating the uncoupling effect of introducing a void layer in the mesh on the system matrix. The nodes are ordered such that the outer shell mesh (Ω_1) comes first and then the inner body mesh (Ω_2)

5.1.4 Including Non-Scattering Sub-Domains

Equation 5.7 has been extensively used for modelling diffusive tissue and for phantom experiments and validation procedures. However it cannot be used to handle non-scattering regions. If we consider the thin layer case in figure 5.1 we can begin to get an idea of the way the RDM will work. Essentially we know we must volume mesh the domain Ω, but our study of graphics (chapter 4) tells us we need only mesh the surface of the void $\overline{\Omega}$.

If we consider that we volume mesh only the diffusive parts of the domain and order the nodes such that we have all the nodes from the outer region together and all those from the inner region together in the system matrix and set-up our standard FEM matrix we will end up with a matrix of the structure shown in figure 5.2. This clearly shows that the outer and inner parts of our diffusive domain have become uncoupled by inserting the void region. In the following section we will describe how we address this problem.

5.1.5 A Non-Local Boundary Condition - modelling $\overline{\Omega}$

To solve the light transport we must somehow re-couple the two parts of the system matrix. If we consider the domain excluding the void then we have a new boundary $\partial\overline{\Omega}$ not included in our original boundary condition, equation 5.5. So we must impose a new boundary condition here, which will couple the two FEM systems, hence it must

be non-local. By non-local we mean that as the surfaces are not necessarily connected together we are forming coupling between disconnected elements, i.e. the coupling isn't local to the nodes.

The boundary condition starts by extending the Robin boundary condition to include an incoming flux.

$$\frac{d}{d\hat{\nu}}\Phi(\mathbf{m}) + \frac{1}{2\alpha\kappa(\mathbf{m})}\Phi(\mathbf{m}) = \Gamma_{vd}(\mathbf{m}) \tag{5.13}$$

where $\Gamma_{vd}(\mathbf{m})$ is some flux incoming at the surface from the void region $\overline{\Omega}$, given by:

$$\Gamma_{vd}(\mathbf{m}) = \int_{2\pi} \phi_{in}(\mathbf{m}, \hat{\mathbf{s}}) d\hat{\mathbf{s}} \tag{5.14}$$

We already know how to transform this integral to a boundary based on one from Radiosity theory (see section 4.3, equations 4.3-4.7), so we can simply write:

$$\Gamma_{vd}(\mathbf{m}) = \int_{2\pi} \Phi(\mathbf{m}') \frac{cos(\eta)cos(\eta')}{\pi|\mathbf{m}, \mathbf{m}'|^2} \mathrm{Vis}(\mathbf{m}, \mathbf{m}') d\mathbf{m}' \tag{5.15}$$

This leads to an equation for the boundary condition of the void regions:

$$\frac{d}{d\hat{\nu}}\Phi(\mathbf{m}) + \frac{1}{2\alpha\kappa(\mathbf{m})}\Phi(\mathbf{m}) = \int_{\partial\overline{\Omega}} \Phi(\mathbf{m}') \frac{cos(\eta)cos(\eta')}{\pi|\mathbf{m}, \mathbf{m}'|^2} \mathrm{Vis}(\mathbf{m}, \mathbf{m}') d\mathbf{m}' \tag{5.16}$$

There is however a slight modification which is required at this point. We have specified our void regions to be non-scattering, however we have made no constraint upon absorption.Therefore we must modify our expression for $\Gamma_{vd}(\mathbf{m})$ to account for this. To do this we add the attenuation term for light travelling along a straight line, giving us a modification by $e^{-(\mu_a(\overline{\Omega})|\mathbf{m},\mathbf{m}'|)}$. This leads us to the complete Radiosity Boundary Condition (RaBC):

$$\frac{d}{d\hat{\nu}}\Phi(\mathbf{m}) + \frac{1}{2\alpha\kappa(\mathbf{m})}\Phi(\mathbf{m}) = \int_{\partial\overline{\Omega}} \Phi(\mathbf{m}') \frac{cos(\eta)cos(\eta')}{\pi|\mathbf{m}, \mathbf{m}'|^2} \mathrm{Vis}(\mathbf{m}, \mathbf{m}') e^{-(\mu_a(\overline{\Omega})|\mathbf{m},\mathbf{m}'|)} d\mathbf{m}' \tag{5.17}$$

Thus our modified version of equation 5.6 becomes:

$$\begin{aligned}\int_\Omega \kappa(\mathbf{r})\nabla\Psi(\mathbf{r})\nabla\Phi(\mathbf{r})d\mathbf{r} + \int_\Omega \Psi(\mathbf{r})\mu_a(\mathbf{r})\Phi(\mathbf{r})d\mathbf{r} + \int_{\partial\Omega} \frac{1}{2\alpha}\Psi(\mathbf{m})\Phi(\mathbf{m})d(\mathbf{m}) \\ - \int_{\partial\overline{\Omega}} \Psi(\mathbf{m}) \int_{\partial\overline{\Omega}} \Phi(\mathbf{m}') \frac{cos(\eta)cos(\eta')}{\pi|\mathbf{m}, \mathbf{m}'|^2} \mathrm{Vis}(\mathbf{m}, \mathbf{m}') e^{-(\mu_a(\overline{\Omega})|\mathbf{m},\mathbf{m}'|} d\mathbf{m}' d\mathbf{m} \\ = \int_\Omega \Psi(\mathbf{r}) q_0(\mathbf{r}) d\mathbf{r}. \end{aligned} \tag{5.18}$$

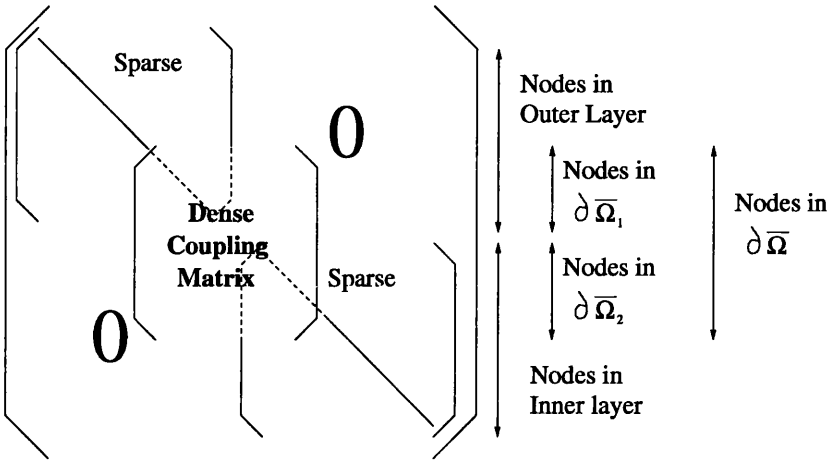


Figure 5.3: A diagram illustrating the effect of the RaBC in re-coupling the FEM matrices of the DA across the void inclusion.

Creating the FEM model for the RDM

Having established our weak formulation of the integral equation for the whole domain Ω including its void inclusions $\overline{\Omega}$ we may now proceed to establish a FEM formulation assuming a Galerkin approximation:

$$(\mathbf{K}(\kappa) + \mathbf{C}(\mu_a) + (\frac{1}{2\alpha}\mathbf{A} - \mathbf{E}))\Phi = \mathbf{Q} \tag{5.19}$$

with matrix/vector elements as before, but extended by:

$$E_{ij} = \frac{1}{2\alpha}\int_{\partial\overline{\Omega}} u_i(\mathbf{m}) \int_{\partial\overline{\Omega}} u_j(\mathbf{m}')f(\mathbf{m},\mathbf{m}')\text{Vis}(\mathbf{m},\mathbf{m}')e^{-\mu_a|\mathbf{m},\mathbf{m}'|}d\mathbf{m}'d\mathbf{m}, \tag{5.20}$$

where $f(\mathbf{m},\mathbf{m}') = \frac{cos(\eta)cos(\eta')}{\pi|\mathbf{m},\mathbf{m}'|^2}$. This leads us to a coupled matrix of the form illustrated in figure 5.3.

5.1.6 Solving the RDM

Having assembled our system of equations we have only to solve for the new E_{ij} elements and we have our solution. This seems simple enough, especially as it has been successful already in 2D [4]. However troubles do arise in the 3D case that are not immediately apparent in a 2D problem. Essentially we have two problems to deal with in assembling matrix **E**, the visibility problem and solving the Form-Factors. Both of these problems experience some non-triviality in their extension to 3D and a selection of models are proposed in chapter 6 to handle these issues. In chapter 7 we will illustrate the effects and relative success of the various models. In the following section we

shall introduce some theoretical models for solving the components of the RDM which we can use as effective benchmarks in developing the RDM.

5.1.7 RDM - Time resolved model

In Optical Tomography it is often useful to be able to handle the time-dependent case. This is because we cannot uniquely recover both absorption and scattering coefficients from just intensity data [5]. Therefore we need to be able to handle the time-dependent case. Also as the readily available optical tomography system, MONSTIR[80], is a time-resolved system, we will need to generate time-resolved data in chapter 7 to validate the RDM.

In order to model the time dependent behaviour of the RDM it is necessary to move the equation to the frequency domain and thus express its temporal moments. The frequency domain DA is given by:

$$-\nabla \cdot \kappa(\mathbf{r})\nabla\Phi(\mathbf{r},\omega) + \mu_a(\mathbf{r})\Phi(\mathbf{r},\omega) - \frac{i\omega}{c}\Phi(\mathbf{r},\omega) = q_0(\mathbf{r},\omega), \tag{5.21}$$

where c is the speed of light in the domain. Which gives us the FEM form:

$$(\mathbf{K}(\kappa) + \mathbf{C}(\mu_a) + \frac{1}{2\alpha}\mathbf{A} + i\omega\mathbf{B})\mathbf{\Phi}(\omega) = \mathbf{Q}(\omega), \tag{5.22}$$

with matrix entries for B given by:

$$B_{ij} = \int_\Omega \frac{1}{c} u_i(\mathbf{r}) u_j(\mathbf{r}) d\mathbf{r} \tag{5.23}$$

To add the Radiosity component we must modify our exponential term to introduce a phase delay so that the entries for matrix E become:

$$E_{ij}(\omega) = \int_{\partial\Omega} u_i(\mathbf{m}) \int_{\partial\Omega} u_j(\mathbf{m}') e^{-(\mu_a(\overline{\Omega})+i\omega/c)|\overrightarrow{\mathbf{m}',\mathbf{m}}|} \frac{(\hat{\mathbf{s}}.\hat{\nu})(\hat{\mathbf{s}}'.\hat{\nu}')}{\pi|\overrightarrow{\mathbf{m},\mathbf{m}'}|^2} \mathsf{Vis}(\mathbf{m},\mathbf{m}') d\mathbf{m}' d\mathbf{m}. \tag{5.24}$$

Thus our modified system matrix for the frequency domain becomes:

$$\mathbf{F}(\omega) = \mathbf{K}(\kappa) + \mathbf{C}(\mu_a) + \frac{1}{2\alpha}(\mathbf{A} - \mathbf{E}(\omega)) + i\omega\mathbf{B}. \tag{5.25}$$

Beginning from,

$$\tilde{F}\mathbf{\Phi} = \mathbf{q_0} \tag{5.26}$$

we obtain the temporal moments by differentiating in frequency:

$$\tilde{F}\boldsymbol{\Phi} = \mathbf{q_0} \tag{5.27}$$

$$\tilde{F}\dot{\boldsymbol{\Phi}} + (B - \dot{E})\boldsymbol{\Phi} = 0 \tag{5.28}$$

$$\tilde{F}\ddot{\boldsymbol{\Phi}} + 2(B - \dot{E})\dot{\boldsymbol{\Phi}} - \ddot{E})\boldsymbol{\Phi} = 0, \tag{5.29}$$

and so on. This system is solved iteratively to yield the moments:

$$\langle \mathbf{t}^n \rangle = \frac{\overset{(n.)}{\boldsymbol{\Phi}}}{\boldsymbol{\Phi}}. \tag{5.30}$$

as shown in [4].

5.2 Analytic Models

In this section we will illustrate some simple analytic models used for testing our model. These are permissible in simplified geometries and allow us to benchmark various components of our code. The circular/spherical models have been developed independently to work presented by Hyvönen in [46], but can be seen to be the same sets of equations (although Hyvönen only presents the 2D case).

5.2.1 Visibility

Visibility is much easier to solve in an analytic sense, it is possible to use geometric objects to construct simple test cases, as we can perform much simpler geometric intersection tests with simple objects such as spheres, cylinders or cubes. As our test case we selected a model we could use easily in validation studies on the whole RDM. The geometry chosen is the concentric sphere geometry as this geometry is easy to simulate in a stochastic model. Another advantage of this geometry is that a far simpler analytic visibility expression can be derived which makes it useful as a benchmark.

Figure 5.4 illustrates the concentric sphere geometry and the spherical coordinates for the relative angle between any two points - (η_{rel}, χ_{rel}) (with respect to the sphere centres). We can consider the visibility problem in this case as a 2D problem as we know that χ_{rel} will not affect the question of visibility between two points (in fact we may also use this visibility model for the 2D concentric circle geometry). This leads us to the geometric arrangement illustrated in figure 5.5, which clearly shows the important visibility events as η_{rel} between two points changes in this case. So we have derived three rules for the visibility in this case:

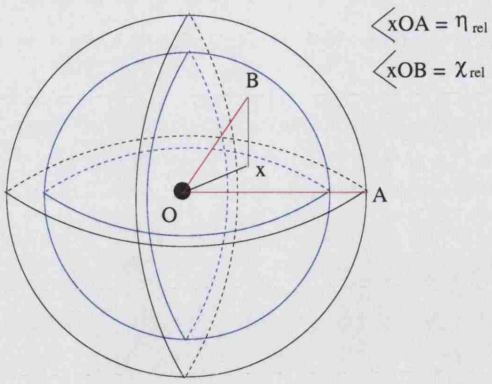


Figure 5.4: A diagram showing the geometry of the 3D concentric sphere problem, showing the relative angles(η_{rel}, χ_{rel}) between two points on the boundary of the domain.

- *Inner sphere to Inner sphere:* Not visible.
- *Inner sphere to Outer Sphere:* Visible if $\eta_{rel} < \eta_1,\ \eta_1 = cos^{-1}(\frac{r}{R})$
- *Outer sphere to Outer Sphere:* Visible if $\eta_{rel} < \eta_2,\ \eta_2 = 2\eta_1$

For future reference we shall refer to η_1 as η_{rco} and η_2 as η_{Rco}. We note that η_{rco} can be considered as describing the visibility horizon and η_{Rco} as describing the visibility cut-off.

5.2.2 Form Factors

In this section we will outline the analytic Form Factors for both the 2D (circular) and 3D (spherical case). As we will be interested in both hollow and clear layer models, we find the analytic expressions for both the hollow outer sphere, and for the case of a concentric layered sphere model.

2D

Consider a circle radius R in parametric co-ordinate (η). The Form Factor at a point A can be expressed in terms of the following equation:

$$\begin{aligned} F_A &= \int_{\eta=0}^{2\pi} g(\eta) R d\eta \\ g(\eta) &= \frac{cos\theta cos\theta'}{2d} \end{aligned}$$

where η is the standard parametric representation for a circle. From the definition of the coordinates we can find the outer-to-outer form factor from a given point A ($\eta = 0$)

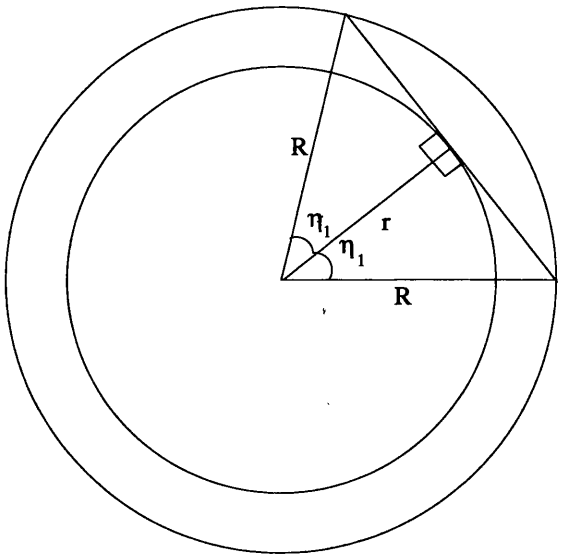


Figure 5.5: A diagram showing the geometry of the visibility events extrapolated to the 2D case based on η_{rel}.

to any other point B (η):

$$
\begin{aligned}
A &= (R, 0) \\
B &= (Rcos(\eta), Rsin(\eta)) \\
ab &= (Rcos(\eta) - R, Rsin(\eta)) \\
d = |ab| &= \sqrt{(Rcos(\eta) - R)^2 + R^2 sin^2(\eta)} = R\sqrt{2 - 2cos(\eta)} \\
\hat{n}_A &= (-1, 0) \\
\hat{n}_B &= (-cos(\eta), -sin(\eta)) \\
cos\theta &= \frac{R - Rcos(\eta)}{R\sqrt{2 - 2cos(\eta)}} \frac{\sqrt{1 - cos(\eta)}}{\sqrt{2}} \\
cos\theta' &= \frac{\sqrt{1 - cos(\eta)}}{\sqrt{2}} \text{from symmetry} \\
g_o(\eta) &= \frac{cos\theta cos\theta'}{2d} = \frac{1 - cos(\eta)}{4R\sqrt{2 - 2cos(\eta)}}
\end{aligned}
$$

A quick check here is provided as if we integrate this function over the domain we get $\int_{\eta=0}^{2\pi} g_o(\eta) R d\eta = 1$, which is correct under the definition of the Form Factor.

Similarly we can find the outer-to-inner Form Factor, simply by expressing point B as a point on an inner circle radius r:

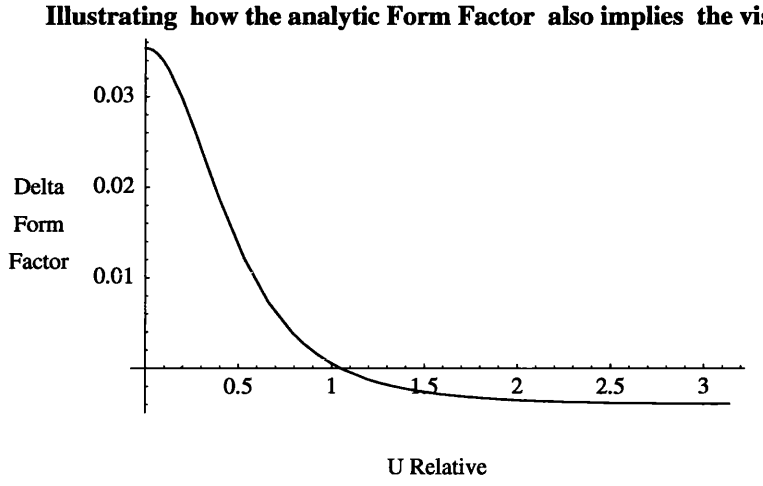


Figure 5.6: A diagram showing the graph of $g_i(\eta)$(y-axis) against η(x-axis) for the case where $R = 5$ and $r = 3$, showing the graph turning negative at η_{rco}

$$
\begin{aligned}
A &= (R, 0) \\
B &= (rcos(\eta), rsin(\eta)) \\
ab &= (rcos(\eta) - R, Rsin(\eta)) \\
d = |ab| &= \sqrt{(rcos(\eta) - R)^2 + r^2 sin^2(\eta)} = \sqrt{r^2 + R^2 - 2rRcos(\eta)} \\
\hat{n}_A &= (-1, 0) \\
\hat{n}_B &= (cos(\eta), sin(\eta)) \\
cos\theta &= \frac{R - rcos(\eta)}{\sqrt{r^2 + R^2 - 2rRcos(\eta)}} \\
cos\theta' &= \frac{-(rcos^2(\eta) - Rcos(\eta)) - rsin^2(\eta)}{\sqrt{r^2 + R^2 - 2rRcos(\eta)}} \\
&= \frac{Rcos(\eta) - r}{\sqrt{r^2 + R^2 - 2rRcos(\eta)}} \\
g_i(\eta) &= \frac{cos\theta cos\theta'}{2d} \\
&= \frac{1 - cos(\eta)}{4R\sqrt{2 - 2cos(\eta)}}
\end{aligned}
$$

This can be checked in the same way as above, except we must take into account the two components of the domain outer and inner and integrate in the visible range:

$$2(\int_{\eta=0}^{\eta_{Rco}} g_o(\eta) R d\eta + \int_{\eta=0}^{\eta_{rco}} g_i(\eta) R d\eta) = 1 \tag{5.31}$$

Interestingly we can also note that in this simple case how in certain circumstances

the Form Factor also expresses the visibility. In the outer-to-outer case we have no such information, but if we look at the graph for the function $g_i(\eta)$ (see figure 5.6) we see that at η_{rco} the function becomes negative, giving a sanity check on our derived analytic models.

3D

Consider a sphere radius R in parametric co-ordinates (η,χ), the Form Factor at a Point A can be expressed in terms of the following equation:

$$\begin{aligned} F_A &= \int_{\chi=0}^{\pi}\int_{\eta=0}^{2\pi} g(\eta,\chi)R^2 sin(\chi)d\eta d\chi \\ g(\eta,\chi) &= \frac{cos\theta cos\theta'}{\pi d^2}, \end{aligned}$$

where η and χ are the standard parametric representation for a sphere, with radius R. From the definition of the co-ordinates we can find the outer-to-outer Form Factor from a given point A ($\eta = 0$, $\chi = 0$) to any other point B (η,χ).

$$\begin{aligned} A &= (0,0,R) \\ B &= (Rsin(\eta)cos(\chi), Rsin(\eta)sin(\chi), Rcos(\eta)) \\ ab &= (Rsin(\eta)cos(\chi), Rsin(\eta)sin(\chi), Rcos(\eta) - R) \\ d = |ab| &= \sqrt{(Rsin(\eta)sin(\chi))^2 + (Rsin(\eta)sin(\chi))^2 + (R - Rcos(\eta))^2} \\ &= R\sqrt{2 - 2cos(\eta)} \\ \hat{n}_A &= (0,0,-1) \\ \hat{n}_B &= (-sin(\eta)cos(\chi), -sin(\eta)sin(\chi), -cos(\eta)) \\ cos\theta &= cos\theta_A = \frac{\hat{n}_A.ab}{|ab|} = \frac{\sqrt{1 - cos(\eta)}}{\sqrt{2}} \\ cos\theta' &= cos\theta_B = \frac{\hat{n}_B.(-ab)}{|ab|} = \frac{\sqrt{1 - cos(\eta)}}{\sqrt{2}} \\ g(\eta,\chi) &= \frac{cos\theta cos\theta'}{\pi d^2} = \frac{1}{4\pi R^2} \end{aligned}$$

Checking the integral again we find $\int_{\chi=0}^{\pi}\int_{\eta=0}^{2\pi} g(\eta,\chi)R^2 sin(\chi)d\eta d\chi = 1$, which satisfies our definition of the Form Factor.

As before (in 2D) we move point B to the inner surface (radius r). This lead us to

the following derivation for the point to point Form Factor.

$$
\begin{aligned}
A &= (0,0,R) \\
B &= (r\sin(\eta)\cos(\chi), r\sin(\eta)\sin(\chi), r\cos(\eta)) \\
\hat{n_A} &= (0,0,-1) \\
\hat{n_B} &= (\sin(\eta)\cos(\chi), \sin(\eta)\sin(\chi), \cos(\eta)) \\
\bar{ab} &= (r\sin(\eta)\cos(\chi), r\sin(\eta)\sin(\chi), r\cos(\eta) - R) \\
d = |ab| &= \sqrt{r^2\sin^2(\eta) + (r\cos(\eta) - R)^2} \\
&= \sqrt{r^2 + R^2 - 2rR\cos(\eta)} \\
\cos(\theta_A) &= \frac{R - r\cos(\eta)}{\sqrt{r^2 + R^2 - 2rR\cos(\eta)}} \\
\cos(\theta_B) &= \frac{-r\sin^2(\eta) - r\cos^2(\eta) + R\cos(\eta)}{\sqrt{r^2 + R^2 - 2rR\cos(\eta)}} \\
&= \frac{R\cos(\eta) - r}{\sqrt{r^2 + R^2 - 2rR\cos(\eta)}} \\
g(\eta,\chi) &= \frac{cos\theta cos\theta'}{\pi d^2} \\
&= \frac{(r - R\cos(\eta))(R - r\cos(\eta))}{\pi(r^2 + R^2 - 2rR\cos(\eta))^2}
\end{aligned}
$$

This clearly shows that as would be expected from the geometric symmetry the Form Factor remains independent of χ, as this is the angle which describes the circular symmetry of the problem. However this part of the Form Factor is dependent on η. Again we can show that $g(\eta,\chi)$ outer-to-inner becomes negative after passing the visibility horizon.

5.2.3 Towards a Self-delta Form Factor Definition

More details of how we generate Form Factors will be seen in the next chapter. However one key issue remains, the self-delta Form Factor, or the Form Factor for a point with itself. This is often unconsidered in graphics as we are dealing with polygonal models, obviously these are not self visible. However the question of self-visibility does arise if we consider the self-visibility or self-delta Form Factor on a curved surface. If we analyse the situation we know that on a convex surface the point is again not self-visible, but on a concave surface it is unclear. Consider then the simplest concave surface - a hollow sphere, if we analyse the delta Form Factor here we begin to get an approach that works.

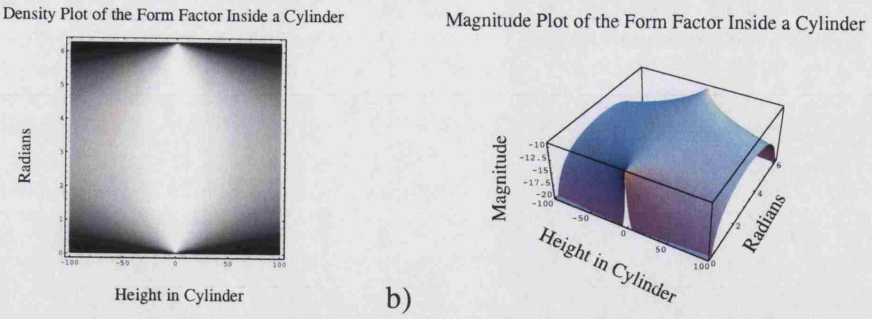


Figure 5.7: Two graphs illustrating the delta Form Factor over the inside of a cylinder

We know from radiant energy exchange that:

$$\int_{\partial\overline{\Omega}} F(\mathbf{m},\mathbf{m}')\mathsf{Vis}(\mathbf{m},\mathbf{m}')d\mathbf{m}' = 1 \forall \mathbf{m} \in \partial\overline{\Omega} \tag{5.32}$$

where $F(\mathbf{m},\mathbf{m}')$ is the form factor between $\mathbf{m}$ and $\mathbf{m}'$. Given this and the expression for the delta Form Factor between two points on the inside of a sphere it is clear that a concave point is self visible.

We must now consider more general cases, to do this we will consider another simple geometry, but a revealing one, the cylinder (η, z) geometry. We chose this as we have varying curvatures, but also any component with the same η but different z will not be mutually visible. This leads to a singular event at a point $\mathbf{m}$ on the cylinder as in the circular plane it should be visible. To verify this we show the delta Form Factor for the cylinder, clearly it mimics the sphere's flat form factor at equivalent z values, but has zero in equivalent η, but varying z (see figure 5.7). Given this we have a singularity at the self-delta Form Factor. We can see that the self-delta Form Factor is essentially varying dependent upon the curvature. In the RDM it has been sufficient to select the largest value of self-delta Form Factor for calculating the integral delta Form Factors. It may be more precise to vary the self-delta Form Factor in a calculation using some tensor representation of its value at a point.

In this work we will use the definition for a self-delta Form Factor $SdFF$ as follows:

$$SdFF(\mathbf{m}) = \frac{1}{4\pi R^2} \text{ where } R = \text{Radius of smallest enclosed sphere at } \mathbf{m} \tag{5.33}$$

5.3 Localised P-refinement

In our experiments we have found that, despite being a computationally more efficient approach to the problem than others [4], the RDM still has a high overhead related to mesh density. It has been found that whilst generally available mesh generation methods do provide good meshes for solving the problem, they can have a very high number of nodes. As a result some problems have proved difficult to compute on available 32-bit hardware. One solution to this problem is to optimise the mesh structure to our individual needs.

It has been found that certain areas of the mesh require denser meshing to provide a stable solution than others. There are a variety of ways (discussed in chapter 2) to address this problem. First is to re-mesh using an h-refinement strategy where we make the elements smaller, this requires re-meshing and also with the mesh tools available is still difficult to create reasonable solutions. Another is to use an r-refinement model where we simply move nodes around to suit our needs. This approach has been tried but tended to lead to skew or skinny tetrahedra in the regions depopulated of nodes. The final approach was to use an element order or p-refinement strategy. This approach removes the need for re-meshing which is a major advantage as this can be a slow and laborious procedure.

There is extensive literature on the subject of p-refinement in FEM, a good background is given in [11]. There is a large selection of literature on the use of p/h-p FEM, however commonly the use of Hermite [110] or Lagrangian [87, 103] bases are used. Currently the existing software uses standard polynomial bases for linear, quadratic and cubic[1] elements, so mixed order elements were developed for these cases.

5.3.1 Deriving Basis functions for the Mixed Order Elements

The basis functions developed were for the set of possible mixed order elements between orders in the existing software. In 2D there are three mixed order elements up to and including third order elements. This is because we may assume that the mixed element will only be of the type with one higher order side as we are using triangular elements. This is because where we have two higher order sides it becomes easier (and more sensible) to use a full higher order element and add a single-sided mixed element.

[1] cubic elements only in $2D$

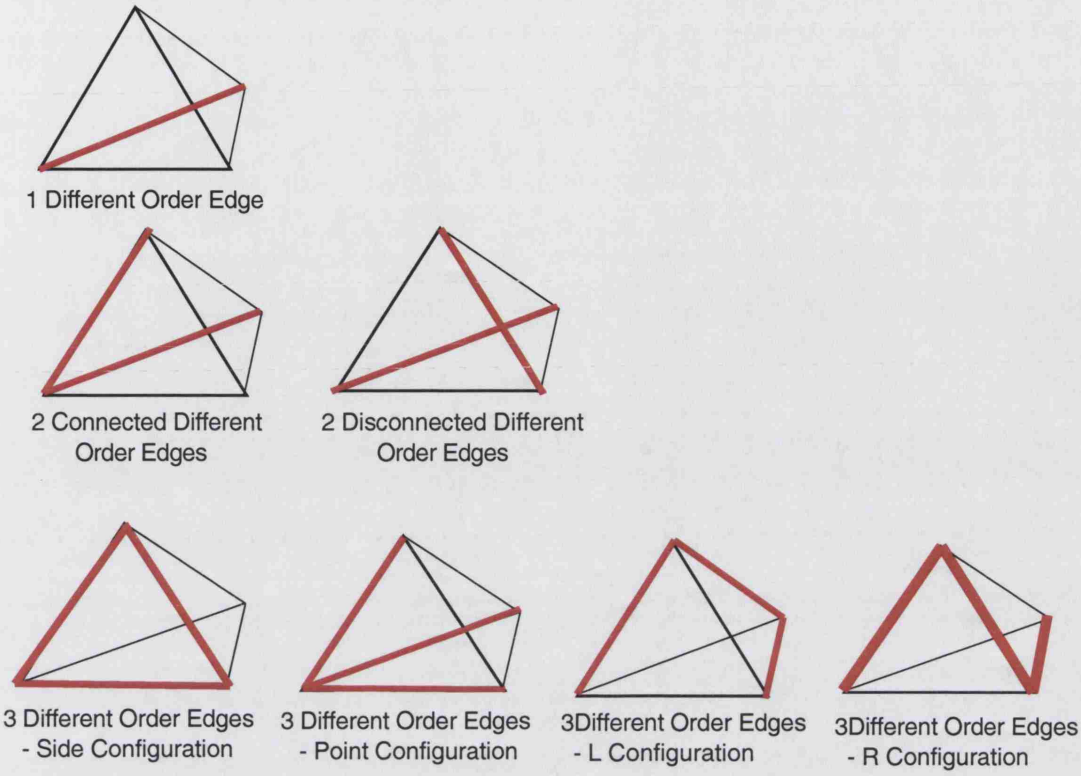


Figure 5.8: A diagram showing the ten possible 3D mixed-order element formations.

In 3D it is more complex as there exist for each mixed type (e.g. linear-quadratic) ten possible permutations of the element (see figure 5.8). Initial tests have been carried out using an implementation with only the higher order 3 edge-side configuration and single edge configuration (as this is possible to mesh given certain constraints on the distribution of higher order nodes).

FEM basis functions were derived for the 2D cases and for the 3D linear-quadratic case. The properties required of the functions are simple:

- Basis functions sum to one across the element.
- Basis functions are one at their given node and zero at all others.
- Basis functions at each edge match the function on the neighbouring element.

The last item in this list is a given for standard symmetric single-order element types, but must be explicit in the mixed-order methods as a requirement.

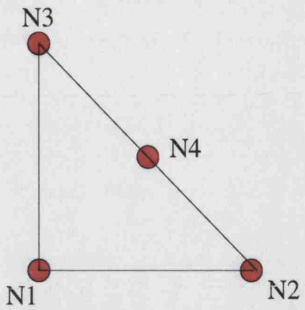


Figure 5.9: A diagram showing the node configuration for the 2D linear-quadratic mixed order p-element.

2D linear-quadratic shape functions

The configuration of this element is shown in diagram 5.9, the node shape functions are as follows:

- Node 1: x = 0, y=0; $f(x, y) = 1 - x - y$
- Node 2: x = 1, y=0; $f(x, y) = x(1 - 2y)$
- Node 3: x = 0, y=1; $f(x, y) = y(1 - 2x)$
- Node 4: x = 1/2, y=1/2; $f(x, y) = 4xy$

2D quadratic-cubic shape functions

In configuring the quadratic to cubic element it was necessary to consider whether to use a 7 or 8 noded triangle. By this we must consider whether to use the internal node which appears in a cubic element, it transpires after investigation that the use of an 8th node makes the generation of correct basis functions much harder, so a 7 node configuration was adopted (see figure 5.10).

The node shape functions of the 7 noded quadratic-cubic element are as follows:

- Node 1: x = 0, y=0; $f(x, y) = 2(1 - x - y)(\frac{1}{2} - x - y)$
- Node 2: x = 1, y=0; $f(x, y) = \frac{-3}{2}x(y - \frac{2}{3})(2x - 1 - y)$
- Node 3: x = 0, y=1; $f(x, y) = \frac{-3}{2}y(x - \frac{2}{3})(2y - 1 - x)$
- Node 4: x = 1/2, y=0; $f(x, y) = \frac{-32}{15}x(1 - x - y)(y - \frac{8}{15})$

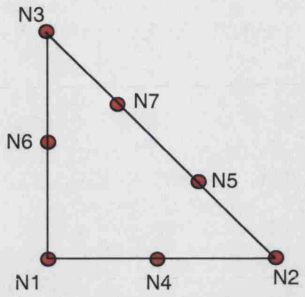


Figure 5.10: A diagram showing the node configuration for the 2D quadratic-cubic mixed order p-element.

- Node 5: x = 2/3, y=1/3; $f(x,y) = \frac{-27}{2}xy(y - \frac{2}{3})$
- Node 6: x = 0, y=1/2; $f(x,y) = \frac{-32}{15}y(1 - x - y)(x - \frac{8}{15})$
- Node 7: x = 1/3, y=2/3; $f(x,y) = \frac{-27}{2}xy(x - \frac{2}{3})$

3D linear-quadratic shape functions side configuration

The configuration of this element is shown in diagram 5.11, the node positions and shape functions are as follows:

- Node 1: x = 0, y=0, z=0; $f(x,y) = (1 - x - y - z)(1 - 2x - 2y - 2z)$
- Node 2: x = 1, y=0, z=0; $f(x,y) = x(2x + 2y - 1)$
- Node 3: x = 0, y=1, z=0; $f(x,y) = y$
- Node 4: x = 0, y=0, z=1; $f(x,y) = z(2z + 2y - 1)$
- Node 5: x = 1/2, y=0, z=0; $f(x,y) = 4x(1 - x - y - z)$
- Node 6: x = 0, y=1/2, z=02; $f(x,y) = 4z(1 - x - y - z)$
- Node 7: x = 1/2, y=1/2, z=0; $f(x,y) = 4zx$

3D linear-quadratic shape functions edge configuration

The configuration of this element is shown in diagram 5.12, the node positions and shape functions are as follows:

- Node 1: x = 0, y=0, z=0; $f(x,y) = (1 - x - y - z)(1 - 2z)$

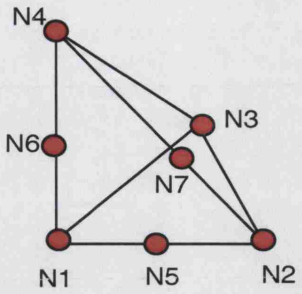


Figure 5.11: A diagram showing the node configuration for the 3D linear-quadratic side configuration p-element.

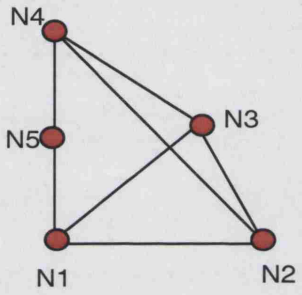


Figure 5.12: A diagram showing the node configuration for the 3D linear-quadratic single edge p-element.

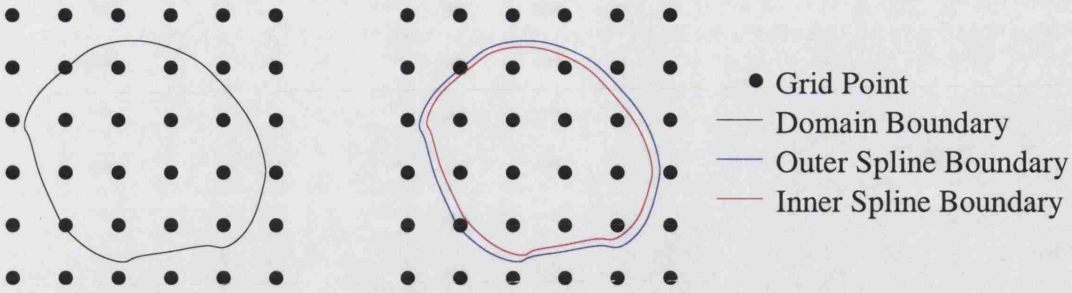
- Node 2: x = 1, y=0, z=0; $f(x, y) = x$
- Node 3: x = 0, y=1, z=0; $f(x, y) = y$
- Node 4: x = 0, y=0, z=1; $f(x, y) = z(2z + 2y + 2x - 1)$
- Node 5: x = 1/2, y=0, z=0; $f(x, y) = 4z(1 - x - y - z)$

5.3.2 Using the localised p-refinement elements within the RDM

Having derived the elements required theoretically they can now simply be inserted into the basic FEM model. As such do not affect the equations of the RDM in any way, in the next chapter we will discuss the implementation issues associated with these new element types.

5.4 A Simple Meshless-FEM Approach to the RDM

In this section we will outline a new approach to solving the RDM. After realising that one of the major problems in the RDM is that of mesh generation several approaches



a) Showing the initial Meshless Grid and the boudary of the real domain

b) Showing the initial Meshless Grid and the two boundaries used to effect the domain with splines

Figure 5.13: A figure illustrating the principles of the meshless FEM approach.

were taken to try to resolve this issue. One idea, already discussed, has been to use a p-refinement strategy to make the process of tetrahedral mesh generation easier. In this section however we will discuss a novel method for solving Finite-Element problems. Here we develop a simplified version of the new Meshless Methods [45, 10] developed to make complex geometries easier to model in FEM codes. First we will briefly outline the workings of a classical Meshless Method, then we will go on to present what is essentially a simplification of this, which we will later show to be a reasonable model for our problem. We will cast our equations here in terms of the RDM as this is the model we are working with, but suggest we could apply this to any FEM problem as the approach is not constrained by our equation but by the principles of the FEM.

5.4.1 The Classical Meshless Approach

The Meshless Finite Element Method, described in [45, 10], initially seems a contradiction. After all we must have elements which therefore implies a mesh. The idea however is to use a grid which does not in itself respect the boundaries of the object (see figure 5.13a). On this mesh we build a basis in terms of splines. This has the advantage of providing a continuous basis over the domain. The next step is to build the boundary into the problem using these basis functions.

The idea involves solving our PDE across the entire grid domain but weighting it with an extra set of basis functions determined by the splines to determine whether a point is inside or outside or on the boundary of our true domain. In order to do this

two boundaries are created, the true boundary and a slightly smaller inner boundary (see figure 5.13b). The spline weighting functions are then set-up to have a zero value outside the outer perimeter and a value of one inside the inner with a smooth change over between the two.

The two main disadvantages of this method are its complex set-up procedure using the spline bases, and the fact that boundary values will tend to be slightly less accurate than the rest of the solution. The advantages of this over a tetrahedral mesh are two-fold. First the actual mesh generation is simple by comparison and second we can directly control the numerical error margin in mesh set-up. This is particularly significant in 3D, as it cannot be done using even the more sophisticated 3D meshing techniques, where error margins can be limited to a certain degree, but must be determined post set-up.

5.4.2 A simplified approach to the Meshless model

Looking at this we decided to look for a simpler approach. As we already have software developed within the group to represent surfaces parametrically and to define surface triangulations from them [106], it was decided to look for a way to exploit these methods. Essentially we are looking at using a voxel model with an associated surface as in the meshless approach, but with certain differences.

Our method uses the idea of using a voxel approximation of the domain to represent the object, but to solve the boundary conditions on the true surface. The principle is that we make certain assumptions in assembling our boundary conditions, here we simply take one more. We assume that given a voxel mesh of reasonable fineness for our domain Ω we can represent its boundary condition not on the boundary of the voxels $\partial\Omega$ but on some approximate associated boundary $\partial\tilde{\Omega}$.

In order to make this idea clearer we will present it here using specific nomenclature with a figure explaining each of the different approximations to the domain. In a standard unstructured approach we would represent a domain Ω with Ω_p^h and its boundary $\partial\Omega$ with $\partial\Omega_p^h$, whilst in a grid approach we would have Ω_v^h and $\partial\Omega_v^h$. Clearly the boundary $\partial\Omega_p^h$ is geometrically closer to $\partial\Omega$ than $\partial\Omega_v^h$, in our meshless method we intend to replace $\partial\Omega_v^h$ with $\partial\Omega_{vm}^h$ which is of the same order of approximation as $\partial\Omega_p^h$, this is illustrated in figure 5.14.

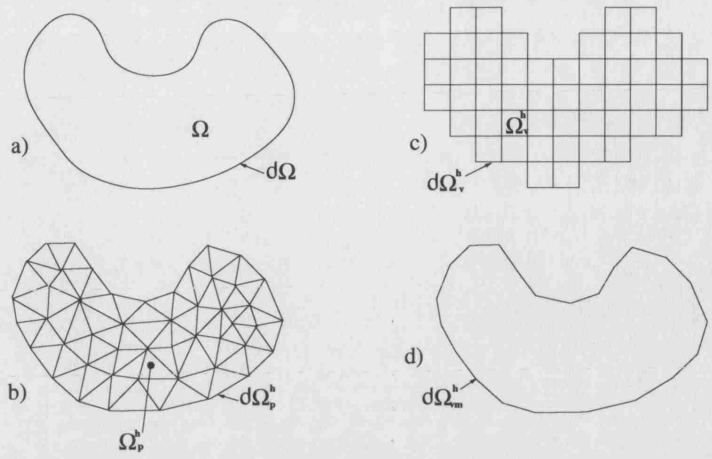


Figure 5.14: A diagram showing (a) our original domain and its boundary (Ω and $\partial\Omega$), (b) a classic FEM representation of the domain and boundary (Ω_p^h and $\partial\Omega_p^h$), (c) a simple pixel FEM grid of the domain and boundary (Ω_v^h and $\partial\Omega_v^h$), and (d) an example of a boundary $\partial\Omega_{vm}^h$ we could associate to domain Ω_v^h in place of boundary $\partial\Omega_v^h$.

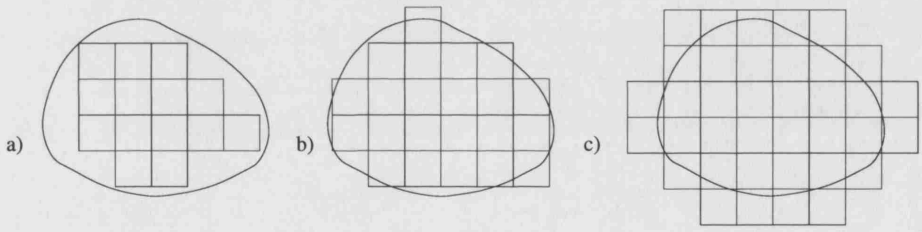


Figure 5.15: A diagram showing the three ways of selecting voxels to be in Ω_v^h.

As we have mentioned one of the advantages of this method is that it is simpler to generate voxel meshes than unstructured ones. The only difficulty here is to decide on the criteria for a voxel being in or out of Ω_v^h, as illustrated in figure 5.15 we have three basic choices:

a) $\tau \in \Omega_v^h \iff (Area(\overline{(\Omega)} \cap \tau) = 0 \&\& Area(\Omega \cap \tau) > 0)$

b) $\tau \in \Omega_v^h \iff Area(\overline{(\Omega)} \cap \tau) < Area(\Omega \cap \tau)$

c) $\tau \in \Omega_v^h \iff Area(\Omega \cap \tau) > 0$

where $\overline{(\Omega)}$ is the set notation for not in Ω and not the void notation $\overline{\Omega}$. In chapter 7 we will illustrate two of these models and illustrate the effects of this choice on the data.

If we examine equation 5.19, we can easily see where the meshless method comes into the equation. If we recast equation 5.19 in terms of an unstructured domain we get:

$$\begin{aligned}\int_{\Omega_p^h} \kappa(\mathbf{r})\nabla\Psi(\mathbf{r})\nabla\Phi(\mathbf{r})d\Omega_p^h \quad &+ \quad \int_{\Omega_p^h} \Psi(\mathbf{r})\mu_a(\mathbf{r})\Phi(\mathbf{r})d\Omega_p^h \\ &+ \quad \int_{\partial\Omega_p^h} \frac{1}{2\alpha}\Psi(\mathbf{r})\Phi(\mathbf{r})d(\partial\Omega_p^h) \\ -\int_{\partial\bar{\Omega}_p^h} \frac{\Psi(\mathbf{m})}{2\alpha}\int_{\partial\bar{\Omega}_p^h} \Phi(\mathbf{m}') \ \mathrm{Vis}(\mathbf{m},\mathbf{m}')e^{-\mu_a(\bar{\Omega}_p^h)|\overrightarrow{\mathbf{m},\mathbf{m}'}|}\frac{(\hat{\mathbf{s}}.\hat{\nu})(\hat{\mathbf{s}}'.\hat{\nu}')}{\pi|\overrightarrow{\mathbf{m},\mathbf{m}'}|^2}d\mathbf{m}'d\mathbf{m} \\ &= \quad \int_{\Omega_p^h} \Psi(\mathbf{r})q_0(\mathbf{r})d\Omega_p^h, \end{aligned} \tag{5.34}$$

whereas in terms of a meshless method, we get:

$$\begin{aligned}\int_{\Omega_v^h} \kappa(\mathbf{r})\nabla\Psi(\mathbf{r})\nabla\Phi(\mathbf{r})d\Omega_v^h \quad &+ \quad \int_{\Omega_v^h} \Psi(\mathbf{r})\mu_a(\mathbf{r})\Phi(\mathbf{r})d\Omega_v^h \\ &+ \quad \int_{\partial\Omega_{vm}^h} \frac{1}{2\alpha}\Psi(\mathbf{r})\Phi(\mathbf{r})d(\partial\Omega_{vm}^h) \\ -\int_{\partial\bar{\Omega}_{vm}^h} \frac{\Psi(\mathbf{m})}{2\alpha}\int_{\partial\bar{\Omega}_{vm}^h} \Phi(\mathbf{m}') \ \mathrm{Vis}(\mathbf{m},\mathbf{m}')e^{-\mu_a(\bar{\Omega}_v^h)|\overrightarrow{\mathbf{m},\mathbf{m}'}|}\frac{(\hat{\mathbf{s}}.\hat{\nu})(\hat{\mathbf{s}}'.\hat{\nu}')}{\pi|\overrightarrow{\mathbf{m},\mathbf{m}'}|^2}d\mathbf{m}'d\mathbf{m} \\ &= \quad \int_{\Omega_v^h} \Psi(\mathbf{r})q_0(\mathbf{r})d\Omega_v^h. \end{aligned} \tag{5.35}$$

We note here that the nodes we have on $\partial\Omega_{vm}^h$ are the same in number and form a one to one association with those on $\partial\Omega_v^h$. This means we still have a system matrix of identical size and fill to the original voxel FEM model, meaning we have acquired no extra unknowns and our problem is of the same complexity to solve. The computational savings with such meshless methods come from two areas. First it is simpler to create the voxel element representation of an object so it will tend to have less vertices, but also even with the same number it requires the storage of less elements. Second in the case of tetrahedral elements we must either store or recalculate on the fly all the integrals for each element as they have different shapes, but with voxel elements we need only calculate and store the values once as they remain the same for all elements. One of the more obvious major advantages of the meshless approach is that it allows us to better match the boundaries of our domain within the RDM in the TOAST software.

5.5 Other RDM type models in the Literature

We have shown how our model has been developed to handle non-scattering regions in diffusive domains. It is not the only way we can handle this situation. In fact there are at least three other approaches suggested in the literature. In this work we will show comparisons with the other major models, showing differences and suggesting explanations. We will also indicate which models we believe to be more correct, and the potential uses of each model based upon the differences between them. In order to do this we will here outline the other similar models which have been developed in tandem with this work.

5.5.1 P_1 - RDM

One interesting note is that if we derive the boundary condition in a fuller sense we end up with a slightly modified equation. Essentially this is the difference between a Diffusion Approximation and a P_1 model. This boundary condition has been developed separately in [73, 74] and in [46]. Here we will briefly present the approach, we note however that at this time under a standard FEM approach we cannot represent this fuller equation and must revert to a BEM or similar model. In fact in 3D a BEM has been implemented within the group by Sikora [90]. We have previously presented some comparisons of this model with the RDM with Sikora [90] and include these results in chapter 7.

In [73, 74] the P_1 RDM boundary condition was presented in 2D and 3D but only implemented in 2D. If we start from a physical definition of the boundary flux incorporating the Fresnel Coefficients as a power balance, in 2D, we end up with the P_1 RDM boundary condition:

$$\Phi(\mathbf{m}) = 2\frac{d\phi(\mathbf{m})}{d\hat{\nu}(\mathbf{m})} + \int_{\partial\overline{\Omega}} \frac{cos(\eta)cos(\eta')e^{-\mu_a|\mathbf{m},\mathbf{m}'|}}{2|\mathbf{m},\mathbf{m}'|} \times \mathrm{Vis}(\mathbf{m},\mathbf{m}') \left[\Phi\left(\mathbf{m}' + 2\frac{d\phi(\mathbf{m}')}{d\hat{\nu}(\mathbf{m}')}\right)\right] d\mathbf{m}' \tag{5.36}$$

In 3D the equation becomes:

$$\Phi(\mathbf{m}) = 2\frac{d\phi(\mathbf{m})}{d\hat{\nu}(\mathbf{m})} + \int_{\partial\overline{\Omega}} \frac{cos(\eta)cos(\eta')e^{-\mu_a|\mathbf{m},\mathbf{m}'|}}{\pi|\mathbf{m},\mathbf{m}'|^2} \times \mathrm{Vis}(\mathbf{m},\mathbf{m}') \left[\Phi\left(\mathbf{m}' + 2\frac{d\phi(\mathbf{m}')}{d\hat{\nu}(\mathbf{m}')}\right)\right] d\mathbf{m}' \tag{5.37}$$

We see a subtle difference here in that the current $\frac{d\phi(\mathbf{m})}{d\hat{\nu}(\mathbf{m})}$ has been included in the boundary condition in a way which is not eliminated by introduction to the DA. This leaves us with a model which cannot be implemented under standard FEM. As such the model is used in a BEM implementation where $\frac{d\phi(\mathbf{m})}{d\hat{\nu}(\mathbf{m})}$ can be explicitly modelled.

5.5.2 Modified Diffusion Model

In work published by Bal et al. [13], a modified diffusion model is presented. The argument here is that as we reduce the thickness of the layer we effectively gradually reduce the non-localisation, so can we come up with a local term which approximates the situation. If we consider the RDM, we see that we indeed have some term $I - E$ where the entries E_{ij} in the matrix are small, i.e. we have some modification to the system which is predominantly affecting the diagonal.

In the Modified Diffusion Model (MDM), we have a simple diffusion term which at the boundary interface. This is added to our normal Diffusion Approximation for the domain giving us an equation:

$$-\nabla \cdot \kappa(\mathbf{r})\nabla\Phi(\mathbf{r}) + \mu_a(\mathbf{r})\Phi(\mathbf{r}) - \kappa_c(\mathbf{g})\Delta_{||}\Phi(\mathbf{g}) = q_0(\mathbf{r}), \tag{5.38}$$

where κ_c is a diffusion coefficient determined by the gap width and absorption and curvature and $\mathbf{g}$ is any point on the interface surface. In the case of the concentric sphere model, $\Delta_{||}$ can be shown to be a Laplace-Beltrami Operator, or equivalent to the second derivative in the tangential direction - which effectively makes sense physically as well as mathematically when we consider our problem. This model only works in the limiting case that the gap width is less than some multiple of the diffusion length in the clear layer, i.e. in thin layers. If we consider this in our test case, the $25-20-17$mm radius spheres, this case is slightly outside the tolerance for the MDM. This is illustrated in the chapter 7 along with other examples where the MDM is sufficient to handle the model under examination.

5.5.3 A low-scatter Diffusion Approximation

It has been noted previously [70] that to use a low-scattering value in the Diffusion Approximation does not work, as would be expected from the breakdown of the governing assumptions. In discussion with the authors of [13], however it has been argued that the curves for a fully diffusive regime and those for a low-scattering level of around

$\mu_s = 1.0 \times 10^{-7}\text{mm}^{-1}$ used to simulate a void in the DA (see figure 7.3 in chapter 7) bracket the true solution. As such it may be possible to provide a low-scattering fit value that would replicate the effects of a void with a lower computational expense, the effectiveness of such a model is illustrated in chapter 7.

5.6 Summary

In this section we have presented the theoretical aspects of the RDM for modelling diffusive domains with non-scattering regions. We have shown its linking of transport and graphics theory, and derived analytic expressions (in restricted geometries) for the key aspects of the model. We have also gone on to discuss theoretical approaches to handling the geometry to improve the computational costs of the model, and finally outlined some other approaches to the domain in question. In the next chapter we will go on to discuss the implementation of the RDM including the development of some novel techniques for handling some aspects of the model.

Chapter 6

The Radiosity Diffusion Model - Practical Issues

In this chapter we will discuss the main contributions we have made to the modelling of light transport in scattering domains containing non-scattering spaces in 3D, in terms of implementation. We will first examine the two major aspects of work involved in developing a functional 3D model for this problem. These are based in the area of the Radiosity component of the RDM, the Form Factor (section 6.1) and the visibility question (section 6.2). Finally in this chapter we will go on to discuss two of the approaches we have developed to trying to make the model computationally more efficient (section 6.3). These relate to methods of modelling the domain with more efficiency, these are the element implementations for the localised p-refinement model and the implementation aspects of the simplified meshless model.

6.1 Form Factor Modelling

The modelling of Form Factors is classically done on the basis of polygon to polygon. In our case we seek to introduce these factors to a numerical model which is point based in its sampling. This provides us with our first problem, in 2D [4] this problem is solved trivially as follows:

- For each node take the midpoint of its adjacent sides and construct a line between them.
- Project this line back to the node.
- Use this line for the calculation of Form Factors.

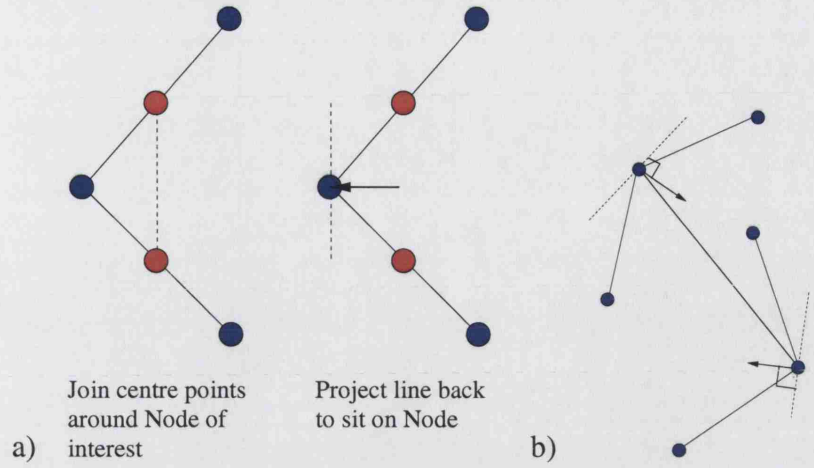


Figure 6.1: Diagrams showing a)Creating the patches for the 2D RDM and b) Using the patches in the D RDM

This process is illustrated in figure 6.1a), and an illustration of its use for Form Factor calculation in 6.1b). In [69] we developed a similar model for the 3D case. This is non-trivial as we cannot simply create a planar patch which fits the midpoints of the sides in a 3D case, see figure 6.2. This problem can also be understood if we consider a set of points $\{\mathbf{x}_i; i = 1, 2, ...X\}$, these are only guaranteed to be coplanar if $X <= 3$. However we can create a set of smaller component polygons as sections of each side connected to the node and conduct a sum over these.

In our work in [69], the approach was to generate a patch around each node by subdividing the triangles of the mesh surface such that each node had support over the part of the triangle nearest it. There are two ways this can be done, first we can simply connect the side centres directly or we can connect them via the edge centres to create the dual mesh to be used for Form Factor calculation (see figure 6.3). If we use the edge centres as well we get a better result as we guarantee the patches being composed of the parts of the sides, and also we will get a more even split of the sides where triangles are slightly skew.

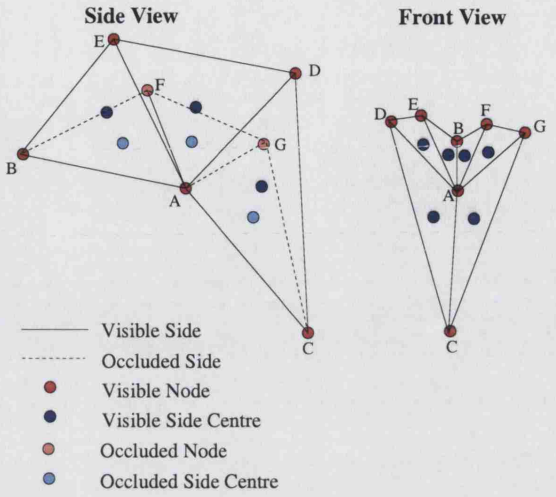


Figure 6.2: A diagram illustrating a side view and a front view of sides around a node in an angled valley geometry - illustrating how the 2D model breaks down in the 3D case.In the side view the occluded objects are those which would be out of sight if the sides were drawn opaquely.

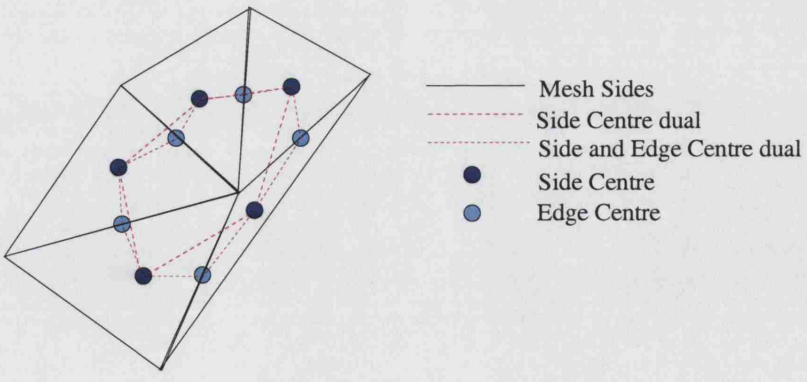


Figure 6.3: A diagram illustrating the choice of dual meshes available in 3D.

Once we have achieved this splitting of the sides we create a Form Factor between the nodes according to the equation:

$$F_{ij} = \sum_{\alpha=1}^{T} \sum_{\alpha'=1}^{T'} F_{\alpha\alpha'} \tag{6.1}$$

where

F_{ij} = the Form Factor between nodes i and j

$\{\alpha = 1, 2, ...T\}$ = the set of side components around node i

$\{\alpha' = 1, 2, ...T'\}$ = the set of side components around node j

$F_{\alpha\alpha'}$ = the Form Factor between side components α and α'

This approach is limited in that it cannot be used for higher order elements without modification and it is based on planar elements, so would be unsuitable in the case of iso-parametric or fully parametric elements. Further it was discovered in [69] that whilst we could generate a reasonable guess of a $3D$ field in a small object, when larger domains were used the model required very fine sampling to produce results, which led to very high computational overheads making the model impractical to use.

Some of the reasons for this can be understood if we have a brief look at how the $2D$ and $3D$ models compare in the analytic case (section 6.1.1). This research has looked at two other approaches which have proved to be computationally cheaper as they can model the problem with lower sampling. We outline these below in sections 6.1.2 and 6.1.3.

6.1.1 Comparison of 2D and 3D behaviour

As we have mentioned a straight usage of the $2D$ model appears to increase our problems computational bounds too highly. If we consider the problem, we might initially perceive that a scaling from $2D$ to $3D$ should give us a simple factor of increase in mesh dimensions. Actually we find that if we look at the shape of the analytic functions in $2D$ and $3D$ for a simple geometry (see figure 6.4) we see that the shape of the underlying function is much steeper in $3D$, which implies in a FEM model that a higher sampling will be required to maintain the same level of accuracy in our model.

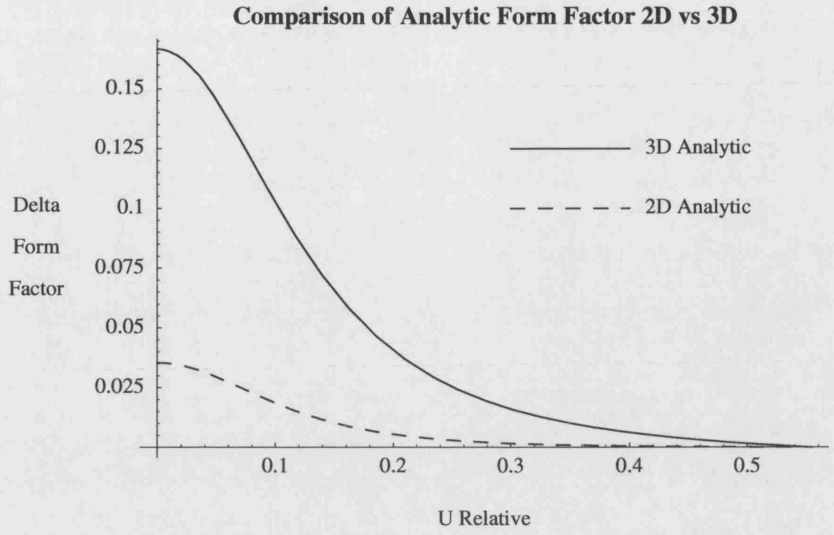


Figure 6.4: A graph showing the difference in rate of change of the Form Factors between a $2D$ and $3D$ analogous case.

6.1.2 Point Collocation

The point collocation method, similar to that developed in [99], can be derived as follows. In this approach we use the traditional radiosity approach of assuming that the sides are small enough such that the Form Factor does not vary across the sides. This leads us to the following equation for the Form Factor:

$$E_{ij} = \frac{1}{2\alpha} \sum_{\alpha=1}^{T} \sum_{\alpha'=1}^{T'} \int_{\tau_\alpha} u_i(\mathbf{m}) \int_{\tau_{\alpha'}} u_j(\mathbf{m}') f_{\alpha\alpha'} e^{-\mu_a |\mathbf{d}_{\alpha,\alpha'}|} \mathsf{Vis}(\mathbf{m}, \mathbf{m}') d\mathbf{m}' d\mathbf{m}, \quad (6.2)$$

where the set of elements τ_α; $\{\alpha = 1, 2, ...T\}$ are those surrounding node i and the set of elements $\tau_{\alpha'}$; $\{\alpha = 1, 2, ...T'\}$ are those surrounding node j (see figure 6.5(a)) and

$$f_{\alpha\alpha'} = \frac{\cos\eta_\alpha \cos\eta_{\alpha'} \mathcal{T}_\alpha}{\pi \left|\mathbf{d}_{\alpha\alpha'}\right|^2},$$

where $\mathbf{d}_{\alpha\alpha'}$ is the vector between the centres of τ_α and $\tau_{\alpha'}$, $\mathcal{T}_\alpha$ is the area of τ_α, $\cos\eta_\alpha = \hat{\mathbf{n}}_\alpha \cdot \hat{\mathbf{d}}_{\alpha\alpha'}$ and $\cos\eta_{\alpha'} = \hat{\mathbf{n}}_{\alpha'} \cdot \hat{\mathbf{d}}_{\alpha\alpha'}$ (as illustrated in figure 6.5(b)). The form of $f_{\alpha\alpha'}$ allows us to simplify equation 6.2 to:

$$E_{ij} = \frac{1}{2\alpha} \sum_{\alpha=1}^{T} \sum_{\alpha'=1}^{T'} f_{\alpha\alpha'} e^{-\mu_a |\mathbf{d}_{\alpha,\alpha'}|} \int_{\tau_\alpha} u_i(\mathbf{m}) d\mathbf{m} \int_{\tau_{\alpha'}} u_j(\mathbf{m}') \mathsf{Vis}(\mathbf{m}, \mathbf{m}') d\mathbf{m}'. \quad (6.3)$$

We can apply two simple visibility methods to this model of the Form Factor. The first is to assume the visibility is constant over the entire support of the basis function of

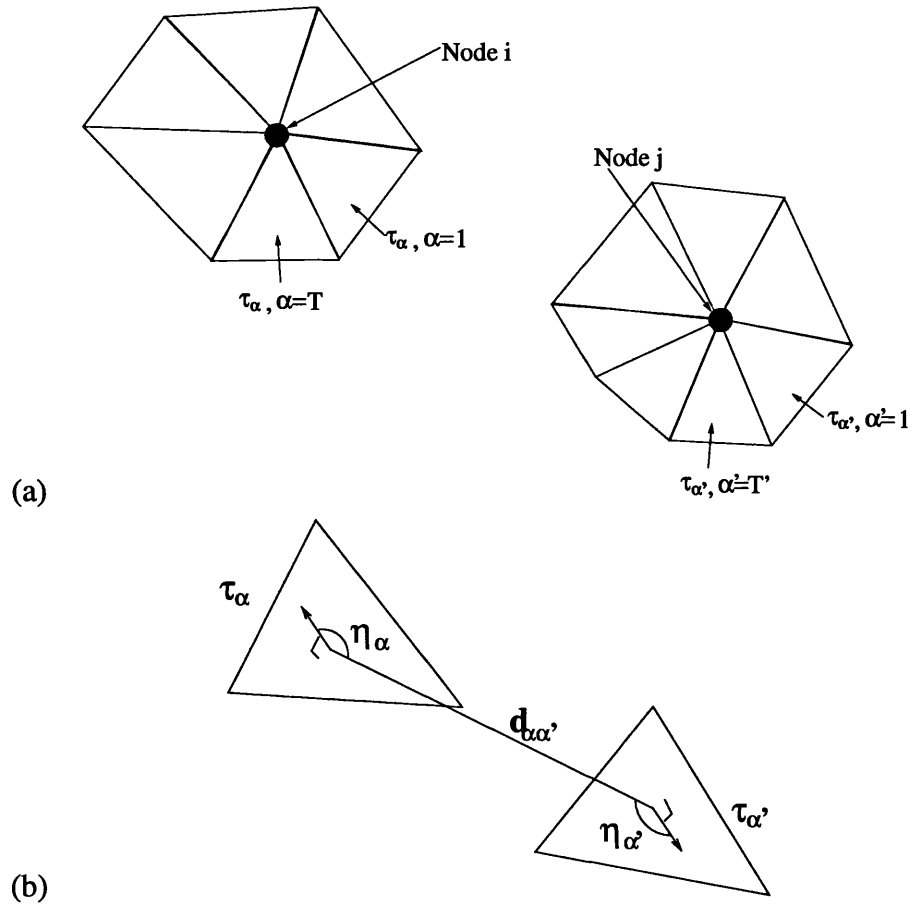


Figure 6.5: A diagram illustrating the geometry and notation of the Side-wise method.

each node. This is obviously naive and may lead us to need adaptive shadow refinement meshing. The other alternative is to use a method like shadow-masking suggested by Zatz [107], which fits into this model by making piecewise visibility tests between each pair of sides in the α and α' sets. This leads us to the following pair of final equations for this method.

Point Collocation - Global visibility

$$E_{ij} = \mathrm{Vis}(i,j)\frac{1}{2\alpha}\sum_{\alpha=1}^{T}\sum_{\alpha'=1}^{T'} f_{\alpha\alpha'}e^{-\mu_a|\mathbf{d}_{\alpha,\alpha'}|}\int_{\tau_\alpha} u_i(\mathbf{m})d\mathbf{m}\int_{\tau_{\alpha'}} u_j(\mathbf{m}')d\mathbf{m}' \qquad (6.4)$$

Point Collocation - Shadow Masking

$$E_{ij} = \frac{1}{2\alpha}\sum_{\alpha=1}^{T}\sum_{\alpha'=1}^{T'} \mathrm{Vis}(\alpha,\alpha') f_{\alpha\alpha'}e^{-\mu_a|\mathbf{d}_{\alpha,\alpha'}|}\int_{\tau_\alpha} u_i(\mathbf{m})d\mathbf{m}\int_{\tau_{\alpha'}} u_j(\mathbf{m}')d\mathbf{m}' \qquad (6.5)$$

where $\mathrm{Vis}(\alpha,\alpha')$ is the visibility at the centre points of α and α'.

6.1.3 Galerkin

Here we change from a piece-wise constant definition of the Form Factor to one varying over the element according to the shape functions. This gives an expression for the Form Factor between two elements as:

$$f_{\alpha\alpha'}(\mathbf{m},\mathbf{m}') = \sum_{k=1}^{N}\sum_{k'=1}^{N'} u_{n(k)}(\mathbf{m})u_{n(k')}(\mathbf{m}')f_{kk'}, \qquad (6.6)$$

where the set of nodes $n(k)$; $\{k = 1,2,...N\}$ are those surrounding node i and the set of elements $n(k')$; $\{k' = 1,2,...N'\}$ are those surrounding node j (see figure 6.5(a)), $n(k)$ maps the local node to the global node to give the correct shape function, and

$$f_{kk'} = \frac{cos(\eta_k)cos(\eta_{k'})}{\pi\,|\mathbf{d}_{kk'}|^2}, \qquad (6.7)$$

where $\mathbf{d}_{kk'}$ is the line between nodes k and k', $\hat{\mathbf{n}}_k$ is the surface normal at k, $\cos\eta_k = \hat{\mathbf{n}}_k.\hat{\mathbf{d}}_{kk'}$ and $\cos\eta_{k'} = \hat{\mathbf{n}}_{k'}.\hat{\mathbf{d}}_{kk'}$ (as illustrated in figure 6.6). Inserting equation 6.6 into equation 6.2 gives us the node to node value as:

$$\begin{aligned} E_{ij} \;=\; & \sum_{\alpha=1}^{T}\sum_{\alpha'=1}^{T'}\sum_{k=1}^{N}\sum_{k'=1}^{N'}\Big[f_{kk'}e^{-\mu_a|r(k,k')|}\zeta_k\int_{\tau_\alpha} u_i(\mathbf{m})u_{n(k)}(\mathbf{m})d\mathbf{m} \\ & \times\int_{\tau_{\alpha'}} u_j(\mathbf{m}')u_{n(k')}(\mathbf{m}')d\mathbf{m}'\Big]. \end{aligned} \qquad (6.8)$$

This method can be split into two methods, as before with the point collocation approach, based on the visibility testing used.

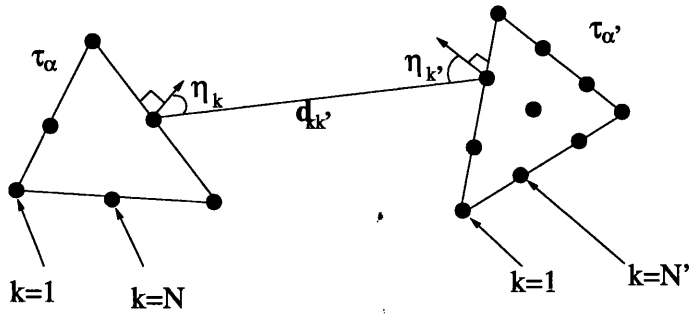


Figure 6.6: A diagram illustrating the geometry and notation of the Node-wise method.

Galerkin - Global Visibility

$$E_{ij} = \mathrm{Vis}(i,j)\sum_{\alpha=1}^{T}\sum_{\alpha'=1}^{T'}\sum_{k=1}^{N}\sum_{k'=1}^{N'}\left[f_{kk'}e^{-\mu_a|r(k,k')|}\zeta_k\int_{\tau_\alpha}u_i(\mathbf{m})u_{n(k)}(\mathbf{m})d\mathbf{m}\right.$$
$$\left.\times\int_{\tau_{\alpha'}}u_j(\mathbf{m}')u_{n(k')}(\mathbf{m}')d\mathbf{m}'\right] \quad (6.9)$$

Galerkin - Shadow Masking

$$E_{ij} = \sum_{\alpha=1}^{T}\sum_{\alpha'=1}^{T'}\sum_{k=1}^{N}\sum_{k'=1}^{N'}\left[\mathrm{Vis}(k,k')f_{kk'}e^{-\mu_a|r(k,k')|}\zeta_k\int_{\tau_\alpha}u_i(\mathbf{m})u_{n(k)}(\mathbf{m})d\mathbf{m}\right.$$
$$\left.\times\int_{\tau_{\alpha'}}u_j(\mathbf{m}')u_{n(k')}(\mathbf{m}')d\mathbf{m}'\right] \quad (6.10)$$

6.1.4 A note on Shadow Masking implementations

When we implement the shadow masking model we must reconstruct the Form Factor matrix structure. This can be seen if we examine figure 6.7, here we see that whilst point G is not visible from point B and therefore not included in our original structure for the form factor matrix, point F is and under equation 6.10 would cause a Form Factor to exist between B and G. This can be implemented by adding entries to the form factor matrix for any node pair which contain neighbouring nodes which are visible.

6.1.5 Other Approaches

As discussed in chapter 4 there are other approaches to modelling the Form Factor which may be more efficient or accurate. The following two sections give outlines of why these approaches have not been implemented in this case. We consider them individually as there are some overlapping reasons, but each case should be considered under its own merits.

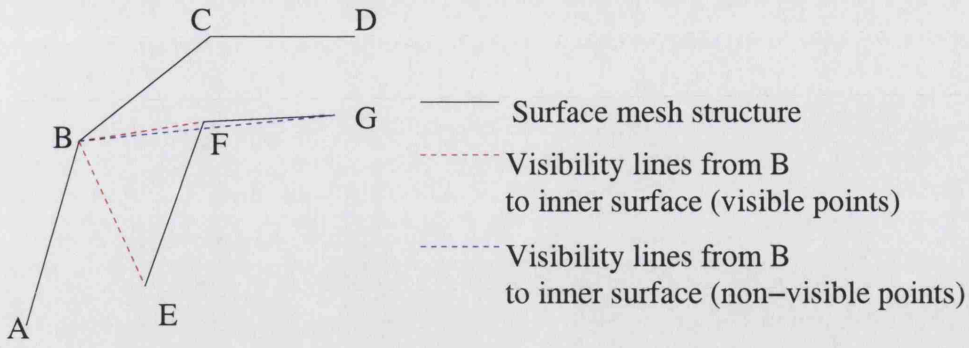


Figure 6.7: A diagram illustrating the structural changes brought about by shadow masking to the Form Factor matrix structure.

Analytic Models

Obviously the best case would be to use an analytic model to solve for the Form Factors. As we are solving a forward problem we can consider our domain known. One argument as to why we might not want to use this is if we were to be reconstructing varying parameters within the void. As this, as yet, is not the case we could consider an analytic model ideal.

We know an analytic model exists for planar surface [84], and that this has been extended to include spherical patches [50], however no generic surface model exists. This provides the first of our problems with using an analytic code as whilst at this time the results we show in this thesis are from standard tetrahedral/voxel meshes (implying a planar model), we can use iso-parametric elements as well and hope to expand to fully parametric models later. Therefore to use an analytic model removes the generic element of our implementation.

The next problem we face is that the model requires that the surfaces be mutually completely co-visible, and in our case this cannot be guaranteed. We could avoid this by simply using a measure where if half or more of the polygons are co-visible we admit them otherwise not, but this to some extent defeats the purpose of the analytic case as our model ceases to be analytic in the global sense.

The third problem is one that may be more easily soluble, in that at the moment the analytic model would only be applicable to the point-collocation method. In that it creates a full patch to patch Form Factor, and we would need to introduce an integration with the basis function to the maths.

We note here that with significant work, probably to the extent of a PhD in its own right we could mitigate the first and third problem in this case. However we are left with the second problem outstanding. We simply cannot get around the co-visibility problem under the methodology suggested in [84]. As such we have used the analytic models as a test for our own models in section 7.2 but have not extended them to be a component of the full RDM.

Hemi-Cube Model

One of the most common models in graphics is the Hemi-Cube model [23]. This allows us to calculate the Form Factors in a way that directly computes the visibility. This initially seems a reasonably good idea. However we have the problem that we are not really working with sides but with nodes. It might be possible to use this to generate the side to side Form Factors requires for the point collocation model, however this removes some of the advantages implied by the one step calculation.

A further note we add here is that this possibility was considered once a full working model had been developed. Initially the investigation of improving the models efficiency was directed at the visibility problem. As will be seen from the results from the OpenGL visibility model developed we see there is a problem with a Hemi-Cube model in our domain and therefore implementation of this algorithm was excluded from our work (see section 7.2.2).

6.2 Visibility Modelling

In this section we deal with the various issues associated with constructing the visibility matrices. We have chosen in this work to focus on three approaches to the visibility, the first is an analytic approach (which we will make no further mention of here as implementation is trivial), given a known geometry we can trivially express the visibility function, the second is a novel approach using the computers hardware to solve the problem for, us and finally we have chosen a spatial discretisation to model the problem.

6.2.1 OpenGL Models

The idea of this approach is to use the on-board graphics processor, common to modern computers, to speed up the visibility calculation. The principle is to use the hardware

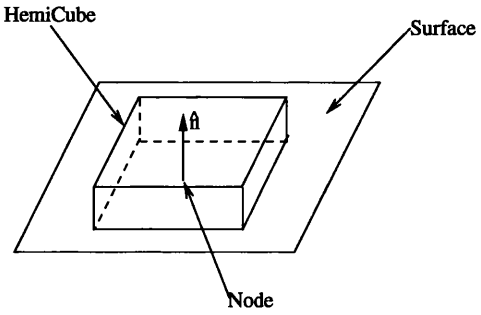


Figure 6.8: A diagram showing the hemi-cube over a node to represent the visible world.

designed for geometric rendering to solve this geometric problem. The OpenGL model has gone through a selection of stages in order to provide a stable model.

The development of the OpenGL hardware algorithm is based upon the hemi-cube radiosity algorithm presented in [23]. The premise of the algorithm is that modern graphics hardware is designed to rapidly render scenes with the correct depth properties, therefore we can generate a map of what is visible from the viewpoint. So provided we can view all the visible space from any point we can see all the points that are visible. In order to create a complete visible map of the world, we place a small hemi-cube over the node to be tested (see figure 6.8) and render each panel of the hemi-cube separately.

A Node Colour Model

The initial idea was to render the scene using white polygons, then to overlay it with the nodes in different colours, by reading back the scene's colour-graph we establish directly which nodes are visible. One constraint we must immediately place on this algorithm is that there are no void objects such that their size is very much smaller than the size of a side in the void. This is because we could end up with a situation as illustrated in figure 6.9, where a large surface area would experience no coupling despite being associated with an energy exchange. This is an aspect of the aliasing problem of the Hemi-Cube algorithm.

In theory, if we perform the rendering of the sides first in the OpenGL model and then set the Z-Buffer to render new points if they are nearer to or equal the current depth at a given point then we should easily read back the set of visible nodes.

Problems and Solutions for the Node Colour Model

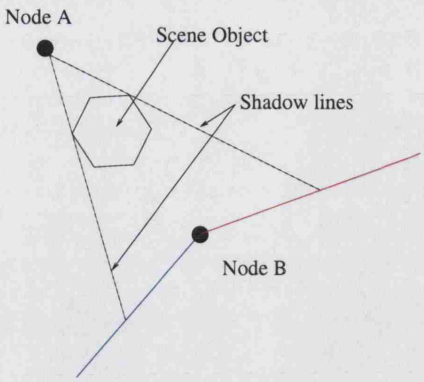


Figure 6.9: A diagram illustrating the size requirements of the OpenGL model.

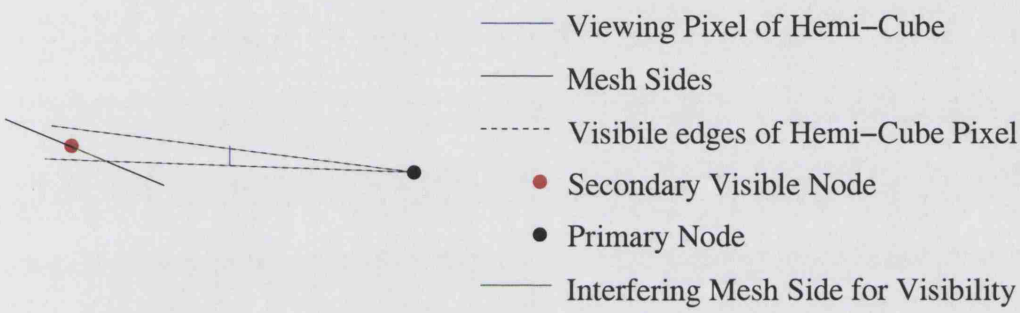


Figure 6.10: A 2D diagram illustrating the problem with using a node-colouring scheme for the OpenGL model .

A problem however occurs with sides which are steeply angled relative to the *primary node* (the node for which we are calculating the visible set). Here we have two problems, first the resolution of the depth buffer may cause a problem and second, we may have the geometry illustrated in figure 6.10. In the geometry case we can see that OpenGL would render the nearest point on the side in the pixel, which, being nearer than the node, would prevent the pixel being viewed.

Two solutions have been considered to the problem caused by the geometry and or depth resolution question. The solutions, however, create their own problems as discussed below. It therefore becomes clear that whilst the OpenGL model is an elegant and rapid solution to the visibility problem in this form its limitations are too severe for our problem. The two solutions are outlined as follows, the first is node lifting, the second node extension.

Node-Lifting: The idea of node lifting is to lift the node up in the visible space. This can be done by two approaches, the first which can be implemented directly or by

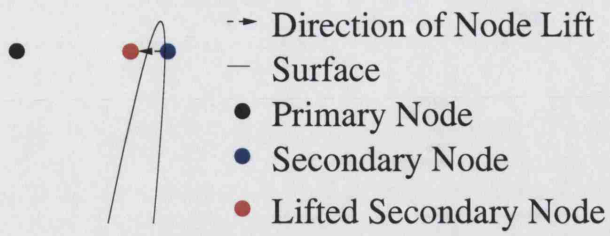


Figure 6.11: A diagram illustrating the visibility errors caused by node lifting approach 1.

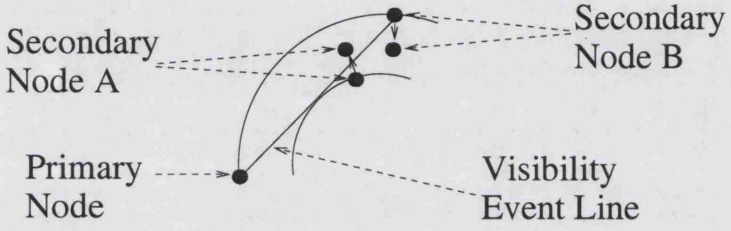


Figure 6.12: A diagram illustrating the two types of visibility errors caused by node extension and by the node lifting approach 2.

setting conditions on the Z-Buffer display decision, is to lift the secondary node along the line between it and the primary node, the problem with this approach is illustrated in figure 6.11. The second approach is to lift the node along its surface normal, this causes the problems illustrated in figure 6.12, where secondary node A appears and B disappears.

Node-Extension: The other approach would be to create a line between the node and its extended point, this can ameliorate the disappearance of node B disappearing, but still allows node A to appear where it shouldn't.

These problems with the Node Colour Model have led to the development of an entirely new approach, the Side Colour Model. This approach is actively different to previous visibility concepts, in that we are not looking at the visibility of a point, but that of the scene around it.

A Side Colour Model

The Side Colour Model is based on the concept of what sides you can see. Here each side is given a different colour, thus if you have a given node associated with a number of sides, you can give it a colour graph based on which sides it is connected to, see figure 6.13. Given this and that we have a colour-graph of visible sides from each node

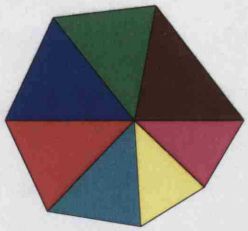


Figure 6.13: A diagram illustrating the concept of the colour graph of a given node in the OpenGL visibility algorithm.

we can easily determine a nodes visibility by the assumption that a node is visible if it's entire colour graph is visible.

An example of how the Side Colour Model works is shown in figure 6.14. It shows the five views from a hemi-cube of a node on the outer surface of a concentric sphere model. Normally in the code we would use a gradual incrementation of colour for each side, regardless of surface, allowing even highly complex scenes to have their visibility determined, in this example we have used a blue scale for the outer sphere and a red scale for the inner sphere to give clarity to the image.

6.2.2 Voxel Visibility

This algorithm is a variation of the hierarchical scene model, it is similar to ideas presented in [2]. The idea is a two stage algorithm, first a partitioning of space into voxels with association of polygons to the partitioning and second a traversal using voxel rays to reduce the number of intersection tests required. The algorithm is as follows:

- Loop 1: for each voxel
  - Inner Loop : for each polygon
    - If side bounding box intersects voxel, mark side as present in the voxel
- Loop 2 : for each node
  - Inner Loop : for each node indexed above that of the outer loop
    - Find first voxel in line between nodes
    - Inner Loop : for each voxel in line
      - Test for intersection with polygons in voxel - stop when intersection found

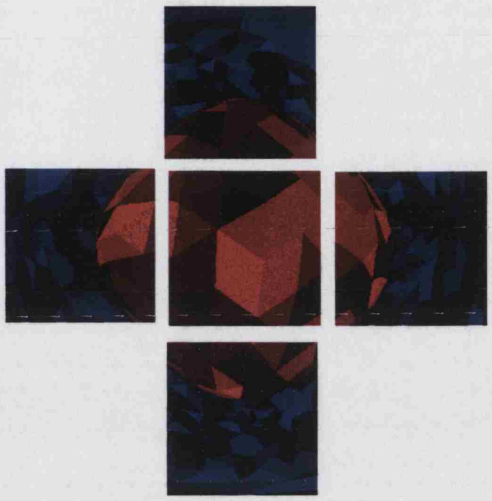


Figure 6.14: A diagram illustrating an example of the five views seen from the hemi-cube in a concentric sphere model. Inner sphere polygons are painted in shades of red and outer sphere polygons in shades of blue to help clarify the image.

    * If no intersection found mark node pair as visible

- Perform self visibility test

Figure 6.15 illustrates the saving effect of the voxel visibility algorithm, it illustrates how only a small proportion of the sides in a concentric sphere model will be tested if we use the voxel algorithm to identify the polygons which might effect the line. One question which this diagram opens is how exactly to choose the size of our voxels, as reducing the voxel size will reduce the number of intersection tests but increase the precomputation time in setting up the voxel mapping.

The speed up obtained in scanning the voxels is obtained by using a compressed row matrix to store the voxel-polygon map, this way we need only check a list of polygons, rather than scanning a whole matrix for marked entries. The other feature we must be careful of here is the self-visibility of nodes, this is determined by checking if at any point the surface around the node is concave.

Figure 6.15: A figure illustrating the saving on resources of the voxel visibility model. Here the voxels are shown on the green line connecting two points, the tested side components are shown in blue with yellow lines for their normals. The small magenta spheres indicate the locations where the various planes of the sides intercept the line, here we clearly have two mutually visible points.

6.3 Geometric Modelling

During the course of this work it has become clear that modelling of geometry is highly important to solving our problem. Firstly we are dealing with a geometric non-local boundary condition, so obviously we must contain sufficient geometric information to represent the problem. The second geometric factor is if we are using meshes which are sufficiently dense to solve the problem we will run into geometric problems with mesh quality - assuming that we use unstructured meshes. In this section we present the practical aspects of the implementation of localised p-refinement and the simplified meshless method we have developed to ameliorate this problem.

6.3.1 Localised P-Refinement - Finding the integrals of the basis functions

The integrals for elements can be computed in two ways. The first is to use quadrature rules, calculating for every different shaped element the integrations, alternatively integration based on natural coordinates can be used to produce a closed form. The latter whilst time-consuming to determine, provides a method for supplying the integral based on the size of the element. It is important to note that if iso-parametric elements are to be used the latter natural coordinate approach is no longer applicable

and quadrature rules must be used instead.

The natural coordinate approach is used in the existing elements and is employed for the mixed-order p-elements. This approach is fairly standard and can be found in most FEM texts (see for example [108]). The integrals are derived from a set of rules based on the natural coordinates of the element. The natural coordinates are in 2D:

$$L_i = \frac{1}{2\Delta}(a_i + b_i x + c_i y + d_i z) \quad , i = 1, 2, 3, 4 \tag{6.11}$$

where:

$$2\Delta = \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix} \tag{6.12}$$

And in 3D:

$$L_i = \frac{1}{6V}(a_i + b_i x + c_i y + d_i z) \quad , i = 1, 2, 3, 4 \tag{6.13}$$

where:

$$6V = \begin{vmatrix} 1 & x_1 & y_1 & z_1 \\ 1 & x_2 & y_2 & z_2 \\ 1 & x_3 & y_3 & z_3 \\ 1 & x_4 & y_4 & z_4 \end{vmatrix} \tag{6.14}$$

The properties of natural coordinates which lead us to the values for the integrals are, in 2D:

$$\nabla L_i = \left(\frac{b_i}{2\Delta}, \frac{c_i}{2\Delta}\right)^T, \tag{6.15}$$

$$\int_A L_1^\alpha L_2^\beta L_3^\gamma dA = \frac{\alpha!\beta!\gamma!}{(\alpha+\beta+\gamma+2)!} 2\Delta \tag{6.16}$$

and

$$\nabla L_i \nabla L_j = \frac{b_i b_j + c_i c_j}{4\Delta^2} \tag{6.17}$$

In 3D these properties are:

$$\nabla L_i = \left(\frac{b_i}{6V}, \frac{c_i}{6V}, \frac{d_i}{6V}\right)^T, \tag{6.18}$$

$$\int_A L_1^\alpha L_2^\beta L_3^\gamma L_4^\eta dA = \frac{\alpha!\beta!\gamma!\eta!}{(\alpha+\beta+\gamma+\eta+3)!} 6V \tag{6.19}$$

and

$$\nabla L_i \nabla L_j = \frac{b_i b_j + c_i c_j + d_i d_j}{36V^2} \tag{6.20}$$

Using the natural coordinates and their properties we can develop any integral of a set of basis functions and their derivatives we chose. From this we obtain a simple expression in terms of the elements nodal coordinates and size for these integrals and these have been derived and implemented for each of our localised p-refinement elements.

6.3.2 Associated Surfaces and Integrals for the Simplified Meshless Method

There are two key features to implementing the Simplified Meshless Method, the integration of basis functions over the surface and the generation of that surface. We will first consider the integration problem as it has a bearing on how we generate our meshes and then the actual process of mesh generation.

Integrating the Boundary Condition

In order to create integrals of the boundary condition we must first decide on a set of basis functions for our surface. These do not necessarily have to be the same as the voxel basis, but obviously they should retain similar support properties in order to keep the generation of the matrix structure simple.

It is clear the easiest basis would be that of the voxels, i.e. a bi-linear basis over the squares, however the lack of co-planarity of four general points in $\mathbb{R}^3$, moves us to a linear triangular basis. Once we have mapped our nodes (and grid) to the true surface then we must create a triangulation of the grid, a trivial problem. Integration of the basis functions then becomes trivial, we only require the integral $\int_\tau u_i(\mathbf{r})u_j(\mathbf{r})d\mathbf{r}$ which is given over triangles as:

$$\begin{aligned}\int_\tau u_i(\mathbf{r})u_j(\mathbf{r})d\mathbf{r} &= \tfrac{1}{6}\mathsf{Area}_\tau \iff i == j \\ &= \tfrac{1}{12}\mathsf{Area}_\tau \quad otherwise \end{aligned} \tag{6.21}$$

So long as we have maintained an association between our voxel mesh and our surface these integrals become trivial for adding the boundary condition on the associated surface.

Creation of the associated surface

One important factor for the Simplified Meshless Method remains to be discussed. That is how we generate our associated surfaces. In this section we will discuss three

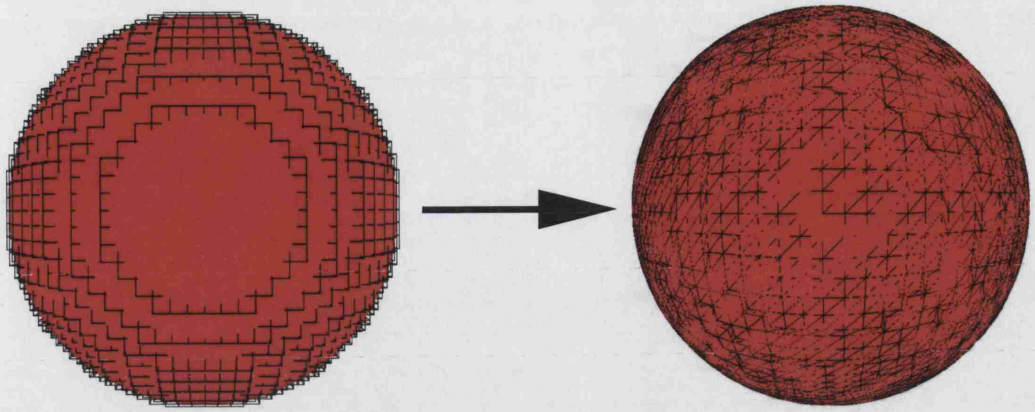


Figure 6.16: A diagram showing how an initial voxel representation of the sphere is mapped to the actual sphere surface.

approaches to this problem:

1) Pushing nodes to a known simple geometry, e.g. a sphere.

2) Relaxing the original grid mesh to form a smooth surface.

3) Using a Voxel to Parametric mapping tool [17, 106] to create a smooth surface.

In the following sections we will briefly outline each method giving an outline algorithm where appropriate in each case.

In the node pushing approach we have started from a simple geometric object such as a sphere or cylinder to generate our voxel mesh. It is therefore easy to project the voxel boundary points to the known true boundary to acquire our associated surface, as is illustrated in figure 6.16. The procedure is as follows:

- Extract the list of surface points to a separate node list - creating a mapping between the two in the process.

- For each point in the duplicate list simply modify its length to be that of the sphere's radius.

- Generate a list of normals being a unit length vector equivalent to the position vector with a sense dependent on the surface (prior knowledge).

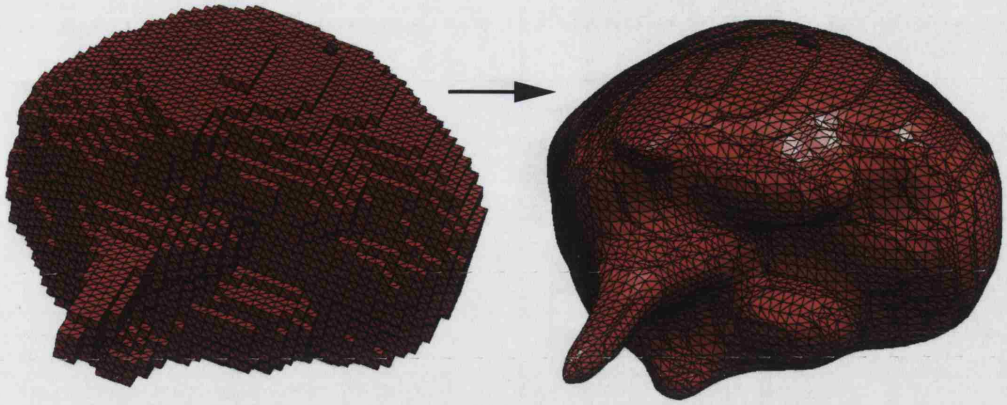


Figure 6.17: A diagram showing the effect of using a simple smoothing algorithm on the surface of a voxel brain taken from an MRI image.

The relaxation method is a simple approach to complex geometries. It relies on a common meshing technique referred to as Laplacian smoothing [14]. The idea is simply to move each to node to the weighted average of its neighbours in an iterative procedure, this leads to a mapping such as that illustrated in figure 6.17.

An algorithm to describe this method is:

- For N iterations do:
  - For each vertex v_i do:
    * $v_{i_{new}} = \frac{\alpha}{J} \sum_{j=1}^{J} v_{j_{old}}$ where v_1 to v_J are the neighbours of v_i and α is some scaling factor.

(We note in [14] the process is a single matrix solve, here we have implemented a simple code, as the speed of this step is not the key issue in our work - this is because meshing has been reduced from hours to minutes and is only done once per object).

The final approach is the most sophisticated and accurate of the three. It involves taking the voxels and creating a parametric representation of the surface in spherical harmonics. This work is being implemented by Zacharopoulos [106] and we will not discuss it here.

6.4 Implementing other Models for Comparison

In this last section we will consider implementation issues associated with the various other models used for comparing to the Radiosity Diffusion Model. First we will mention the codes we have used which have been developed by others, but which have been supplied for comparison. We will move on to mention the implementation of the Modified Diffusion Model[13].

6.4.1 Comparison Models supplied by other Authors

In this work we have used the following codes for comparison:

- Monte-Carlo concentric sphere model - supplied by Eiji Okada at Keio University
- 3D BEM P_1 RDM code - supplied by Jan Sikora at University College London

We would like to thank and acknowledge both for their support in this matter.

6.4.2 Thin-Layers and the Modified Diffusion Model

As we have discussed, a new model for handling thin layers is presented in [13]. In this case an internal boundary condition is implemented to describe the clear region. In order to provide a comparison this model has been implemented for a simple geometry as presented previously by the authors. The idea is to use a tangential integral to represent the light travelling around the clear layer. For the implementation of this model it is necessary to implement the integrals of the form $\int \Delta_{||} u_i(\mathbf{m}) \Delta_{||} u_j(\mathbf{m}) d\partial\Omega$. These integrals are implemented for the spherical case by using Gaussian Quadrature rules on the following form of the Laplace-Beltrami operator:

$$\Delta_{||} u_i \Delta_{||} u_j = \nabla u_i \nabla u_j - \hat{\nu} \cdot (\nabla u_i \nabla u_j) \tag{6.22}$$

It should be possible within the code to implement further integrals to represent more general void geometries, assuming a consistent depth, or even an irregular depth, involving changing the diffusion parameter in equation 5.38. We note that whilst in a simple analytic geometry this diffusion factor can be calculated it is likely that in a more complex geometry of varying thickness that a best fit parameter may be needed.

6.5 Summary

In this chapter we have outlined the more practical aspects of the work we have done in generating the 3D RDM. We have shown all the necessary components to create the

various components of our model and shown the development of various components to the working stage. In the next part of this thesis we will go on to look at the results we have obtained from these models. First validating the RDM, and then analysing its components and the effects non-scattering regions will have on Optical Tomography of the neo-natal head. We will then summarise the implications of this work and potential avenues for development of this model in future work.

Part III

Results and Conclusions

Chapter 7

Results

In this chapter we will illustrate the workings of the RDM. We will first show a validation of the model against a full transport Monte-Carlo code and an initial validation with real data (section 7.1). We will then go on to compare the various models developed for each of the components of the RDM, showing how they compare to each other and the Monte-Carlo simulations as well as to other analytic solutions for individual components of the problem (section 7.2). Next in section 7.3 we will go on to examine the geometric models we have implemented to make our model more efficient, the localised p-refinement model and the Simplified Meshless Method (SMM). Having validated the SMM we will use it in section 7.4 to illustrate the stability of our model under geometric change and also to complete a fuller physical validation from real data. In section 7.5 we will go on to compare the diffusion RDM to other models for our domain type, the P_1-RDM , the MDM and the best-fit low-scattering DA. Finally in section 7.6 we will go on to show the uses of the model in application study of Optical Tomography.

In this chapter three source and detector models will be used [86]. The first is a point source/detector model, here the light source is considered as a delta function one scattering length beneath the surface and the detector is the point response at the surface. The source model here is used as it is common to many different numerical models and is based on the assumption that whilst the DA does not hold near a point directional source, but that it may be considered as an isotropic source once it has travelled one scattering length through a diffusive medium. This source type is used in all the numerical comparisons between different models in this chapter. The second source model used is one which is standardly used for modelling the sources and de-

tectors of the MONSTIR system. Here we consider each source and detector to be a surface distribution in the form of a Gaussian of known standard deviation. this source is also width limited to the same radius as the standard deviation. This source type is chosen as it matches the simple DA phantom data and is considered an effective match for the sources and detectors used on MONSTIR. The final source model we use is used in the blob data simulations, here we use a cosine source detector model. Each source and detector is modelled (again as a surface only function) as having a cosine distribution across the surface, such sources are modelled given a cosine size (the distance equivalent to $\frac{\pi}{2}$) and a width, again the cut-off radius of the source. This source type is used as it provides the most stable fields and allows us to use simpler meshes to provide accurate data responses from the models.

7.1 Validation of the RDM

In order to validate the Radiosity Diffusion Model we have taken three stages. First can we produce believable visual data on the surface of our objects, that is to say is the effect produced by our model of the clear layer physically realistic when we consider the effect we expect to see. If we have a believable model we can then validate it in comparison to other numerical solutions to the transport equation. Finally if the model proves to be numerically equivalent to other full transport simulations, does it compare to real physical data measured on a known phantom using MONSTIR [80].

The initial spherical model was chosen for two reasons, first it fits with simple analytic models developed for test purposes, and second it is trivial to generate a Monte-Carlo model for spherical domains. (For the Monte-Carlo data presented on concentric spheres we must thank Eiji Okada of Keio University Japan.) Our test domain is a set of concentric spheres, the spheres have an outer radius of 25mm, an outer void radius of 20mm and a varying inner void radius (figure 7.1 shows a cut-away section through a typical mesh for this problem). A typical mesh (used for the $25-20-17$mm sphere case has around 15000 nodes, with roughly 4500 void boundary nodes, and the node spacing has been chosen as even (after many experiments this has been found to be the best overall case). In terms of computational cost we need to bear in mind that in a 3mm gap we have that the system matrix will be roughly 25% dense, allowing us reasonable computational size in this case provided we use sparse matrix structures such

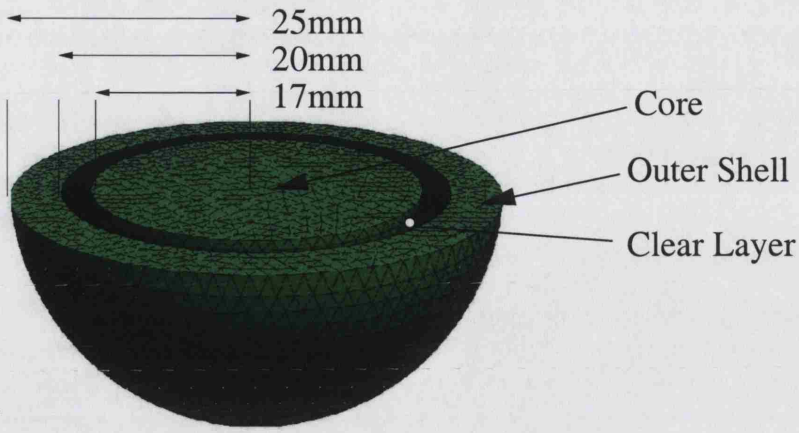


Figure 7.1: A cut-through view of the mesh structure for the spherical shell mesh, illustrating the shell-like void layer used in the test model.

as the compressed row format.We note that changing the size of the mesh will result in a scaling of the number of nodes proportional to the volume change, however that reducing the gap width in proportion to the volume will increase the relative number of nodes needed, and broadening reduces the number. On average it is found that as long as node separations exist which are proportional to the inverse of the scattering coefficient this is usually sufficent in the diffuse media and for the relative gao width in the $25 - 20 - 17$mm model. This provides us with a clear layer model analogous to the effect of the sub-arachnoid space.

In our test models we will use a standard set of parameters, these have been chosen to match some of the typical parameters used in phantom experiments with MONSTiR, they are chosen as they are close to optical parameters measured in tissue [41]. The parameters are as follows, $\mu_a = 0.01\text{mm}^{-1}$, $\mu_s = 1.0\text{mm}^{-1}$ in the diffusive regions and zero in the void (it is actually set at $\mu_s = 10^{-7}\text{mm}^{-1}$ in the Monte-Carlo simulation as it must take some value).

7.1.1 Visual Validation

Our first experiment with the RDM is to check that it is visually producing the correct effect in the data. In other words we wish to ensure that we have a physically believable model before going on to more time-consuming and accurate tests of the models

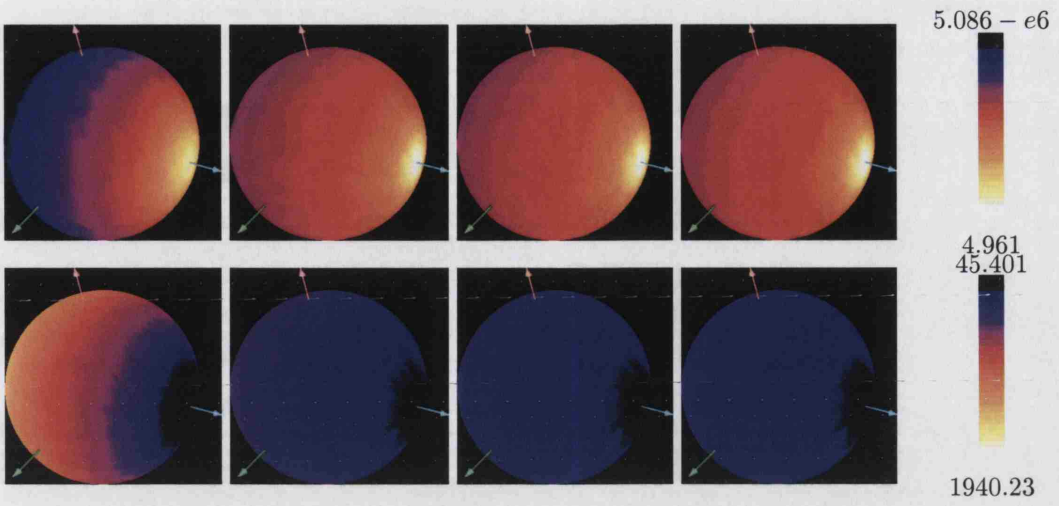


Figure 7.2: Distribution of photon density (log scale), photons/mm^3 (top row) and mean time of photon flight, picoseconds (bottom row) over sphere surface. Left to right, solid sphere (no gap), gap widths 3mm, 4mm, 5mm.

validity. If we look at the effect of the clear layer in physical terms we are effectively reducing the path length of a photon to travel through the medium until it exits. If the absorption remains constant then two effects should be noticed, more photons returning to the surface and a shorter time of flight for these photons.

In figure 7.2 we illustrate the effects generated by the Radiosity Diffusion Model for three gap (3mm, 4mm and 5mm) widths and compare them with that generated for a solid diffusing sphere with data generated by the standard Diffusion Approximation (which we previously showed in [71]). We show both an increase in surface intensity and a reduction in mean time of flight of photons as predicted.

This shows our model to be reasonable in terms of the basic physical effect. We can say the RDM is producing reasonable qualitative results for the introduction of a non-scattering space within a diffusing domain. This allows us to progress with more confidence to examining whether this model is numerically accurate and therefore valid as an approximation of the RTE for the domains we are interested in modelling.

7.1.2 Numerical Validation

The next stage of the validation of the RDM is to compare it with other numerical simulations. In this case we have chosen to use a Monte-Carlo simulation of the RTE. This is chosen as whilst it can be time-consuming to get a stable result it is a proven approach to acquiring reliable results [63]. There are two objectives of this comparison, first to

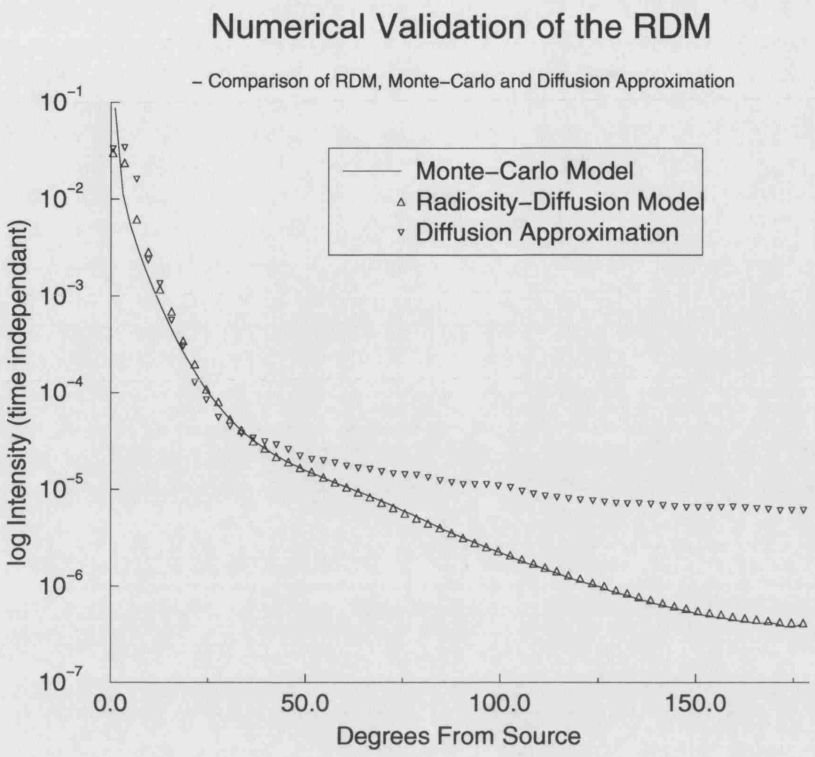


Figure 7.3: A graph illustrating the boundary field generated by Monte-Carlo simulation, the Radiosity Diffusion Model and a simple Diffusion Approximation in a standard concentric spherical void model.

show that the RDM does in fact agree with the solution of the full RTE, validating it as an approximation, but also to show that if we just use a standard Diffusion Approximation with a near zero scattering value ($\mu_s = 1.0 \times 10^{-7}\text{mm}^{-1}$) that it fails to agree with the Monte-Carlo simulation. This would show that the RDM has a place as an approximation to the RTE for modelling domains which are diffusing with non-scattering sub-domains within them. In figure 7.3, we show a logarithmic plot of intensity at the boundary of our standard spherical model with a 3mm clear region. We compare the values produced by the RDM, Monte-Carlo simulation and the DA, against the angular separation from the source. We can clearly see that the RDM does match very well to the Monte-Carlo data, we also see that the Diffusion Approximation fails to match this data, in fact whilst the RDM matches very closely, once we move away from the source the Diffusion Approximation fails to match the data by more than an order of magnitude.

These results indicate two things, first we have shown that the RDM is a valid

approximation to the RTE for our chosen domains. Second we have shown that the DA cannot, as anticipated, handle such domains given a realistic scattering value to the non-scattering space (we note that we cannot assign a value of zero to these sub-domains as this makes the DA meaningless). It has been asked on viewing this graph, whether for a topographic measure we would need a void model as the early region seems to match in the DA case. We point out that whilst the distance to the failure here is probably larger than in the topographic case, the actual differences start to occur in the region where we are starting to sample the data from the clear region and beneath. As Topographic studies aim to measure effects at the surface of the cortex we can state that this is not the case and we will likely need a void model for Topography as well. So at this point we can state that the addition of the Radiosity boundary condition for non-scattering spaces in 3D has successfully extended the range of domains of the RTE which we can approximate with a diffusion based model.

7.1.3 Field Considerations

In this section we consider another important aspect of the RDM and its comparison to the Diffusion Approximation, that is the appearance of the internal fields. If we consider that in Optical Tomography we can show that the boundary data is sensitive to changes based on the product of the source field and the adjoint field from the detector, we must consider how the failure to model the data behaves internally. If the internal field is essentially similar to the RDM for a DA, it may be possible to reconstruct qualitative images, if not quantitative ones in the presence of a void using the DA with no need for a more accurate model.

In figure 7.4 we illustrate the internal fields of the RDM and the DA, showing how the two internal fields differ radically from each other. We see a large transport of light through the clear region resulting in a much altered field in the internal region. This tells us that if we were to use a low-scattering DA to reconstruct from data beneath a clear layer we would fail to obtain meaningful results from the reconstruction. It is in fact one problem of optical tomography that whatever model we chose for the forward problem we will always obtain an image, as such, attempts to use the wrong model may lead to inaccurate interpretations of the data.

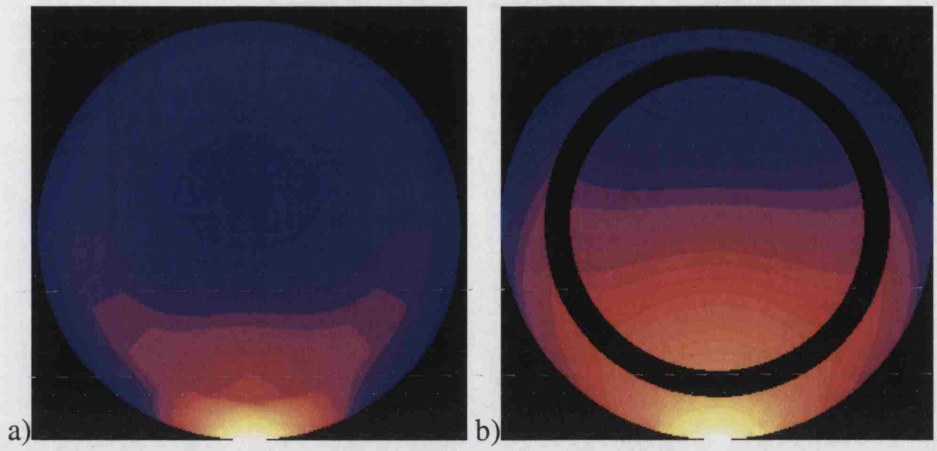


Figure 7.4: a) Plot of the log intensity from a single point source for the xy-plane of the void-sphere derived using the Diffusion Approximation. b) Photon density plot for the xy-plane of the void-sphere derived using the 3D Radiosity-Diffusion model.

7.1.4 Physical Validation

In this section we will show a set of experiments designed to validate the RDM against actual light transport situations. These experiments have been carried out using the experimental set-up MONSTIR at UCL [80] and a selection of phantoms to simulate problems from our domain. We note that there are a wide range of systems available in OT, however as this thesis is concerned only with modelling we have used the hardware present in our institute for this purpose.

In the first set of experiments conducted on void modelling, a glass chemistry flask was used to simulate the void. The standard MONSTIR phantoms are made using an epoxy resin mixed with titanium oxide for scattering and a dye for absorption [35]. The standard set-up is to use a hollow cylindrical resin case and fill this with intra-lipid [29] and then use further resin blobs to simulate optical features. In our case we simply inserted the round bottom flask (shown in figure 7.5) into the cylinder before adding the intra-lipid. Having determined the physical set-up it was also necessary to construct a mesh for the phantom, this was done using NETGEN [81, 82] and the mesh used is shown in a cut-away section in figure 7.6.

Measurements were made on a number of phantoms using a 32 channel time-resolved optical tomography system developed at UCL (MONSTIR). Picosecond

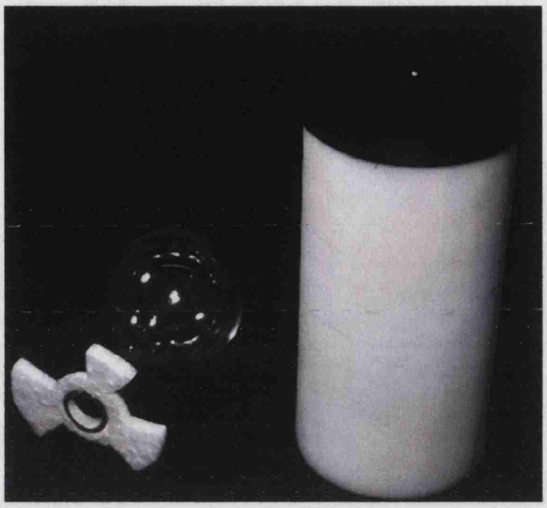


Figure 7.5: A photograph showing the resin cylinder and the round bottomed flask used to simulate the void in the initial physical validation experiments.

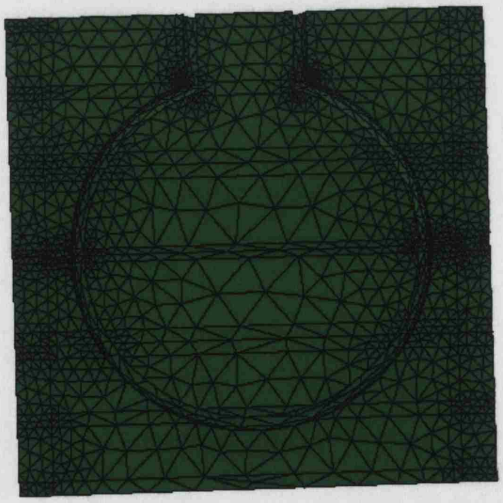


Figure 7.6: A cut through section showing a mesh of the flask experiment (generated using NETGEN[81].

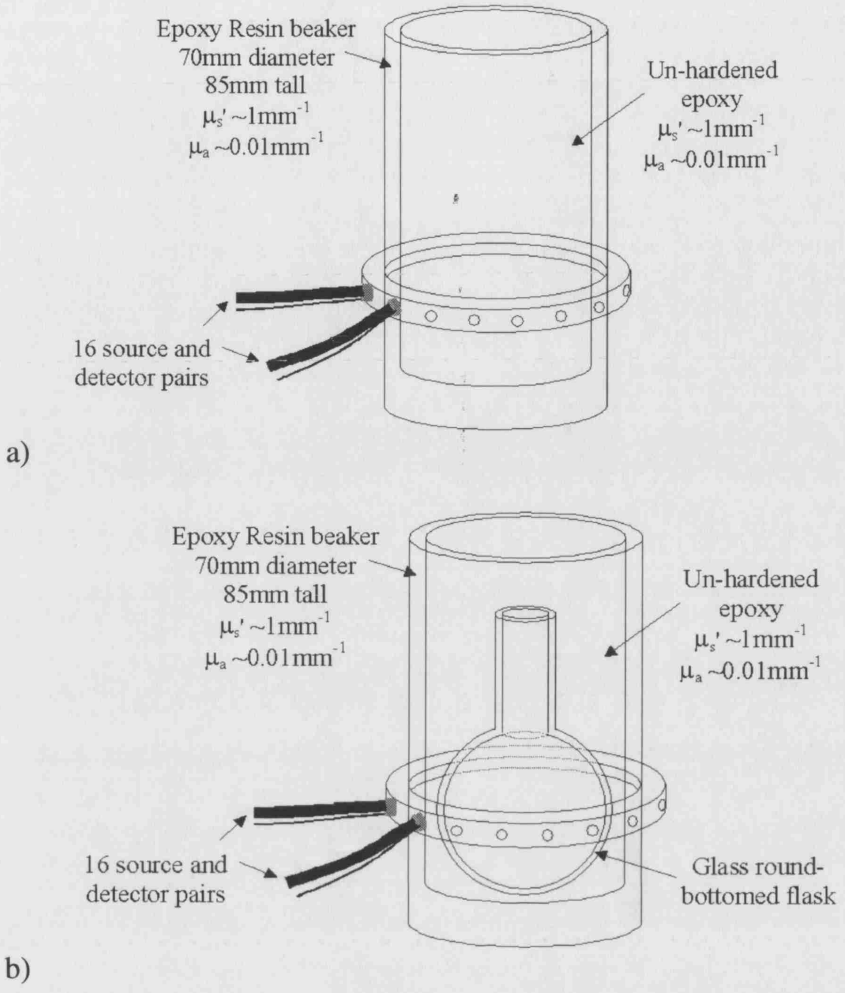


Figure 7.7: a) The experimental set-up for the homogeneous diffusing medium case. b) The experimental set-up for the two clear region experiments.

780nm laser pulses are incident on the phantom. The emerging broadened pulses are measured using time correlated single photon counting, and are known as temporal point spread functions (TPSF's). 16 of the system's sources and detectors were used, positioned in a planar ring as shown in fig 7.7. Each source is sequentially illuminated while detectors very close to the source are deactivated to avoid detector saturation.

Calibration measurements were made to eliminate the differing temporal characteristics of each source and detector. The mean flight time (1st temporal moment) of each TPSF was then extracted. Since the phantoms are circularly symmetric, the measured mean-times were then averaged for each source-detector spacing.

The initial results for the phantom experiments with intra-lipid are shown in figure 7.8. Here we see a comparison of the mean-time of flight of the photons from a source

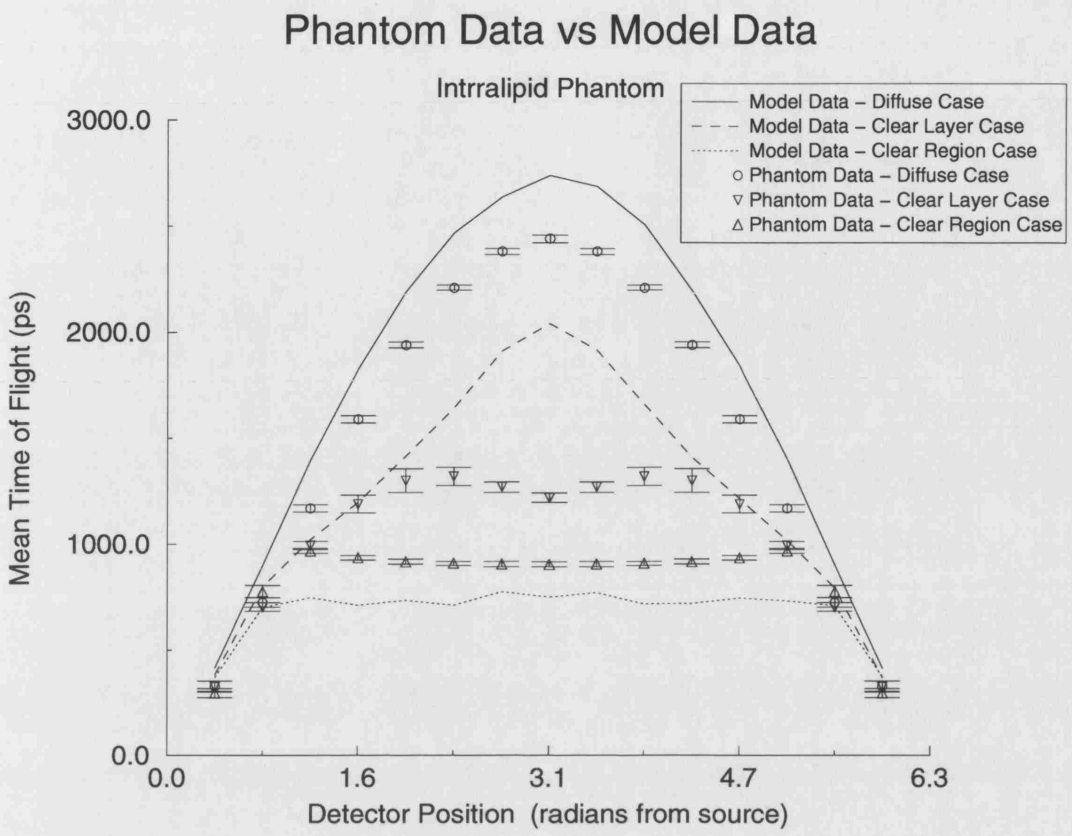


Figure 7.8: A graph showing the comparison of mean-time of flight for the RDM and a resin, glass and intra-lipid phantom.

to the detectors around the periphery of the flask. A comparison of the three data sets, solid diffusing, clear layer and hollow indicates a poor agreement for the non-scattering inclusions. This disagreement could be related to a number of issues, poor match of the physical parameters, a poor geometric match between phantom and mesh or finally to do with the absence of a correct refractive index mismatch model in the RDM. A mismatch of the parameters is unlikely, as while they may not be exact, we should have accuracy to with a few percent from the physical preparations, such a small error would not produce such strong disagreement. The next possibility is the geometry, we know that the flask itself is 2mm thick at the neck, but it may be thinner lower down as it is not necessarily a precise piece of glassware. This again however should not cause such a problem in the data. Finally we consider the refractive index mismatch, it is the most likely candidate and also the easiest to test.

We know that our resin has a refractive index of around 1.56, glass around 1.5

and the intra-lipid somewhere close to 1.4. So the main mismatch will be between the intra-lipid and the other materials. It is noted that currently we use a two part epoxy to generate the resin cylinders, if however we do not include the hardener component of the resin we can generate a fluid material which is of the same optical properties as the solid resin. In doing this we largely, though not completely remove the refractive index mismatch of the phantom.

In figure 7.9, we illustrate how using resin in place of the intra-lipid improves the results, these results were first published in [70]. The results indicate a good match in the data, still not exact, but with the refractive index still not matched exactly this is as we expect.

At this time it was decided to do a fully matched experiment using a resin clear layer, however due to geometry considerations the experiment was not practical computationally with the original RDM. Later in this chapter we will show validation results for a simple meshless model for the RDM (section 7.3.2) which will allow us to model the geometry for a full resin experiment, results of this are shown in section 7.4.2.

7.2 Model Comparisons I - components of the RDM

In this section we will analyse the various methods of calculating Form Factors and visibility. We will compare them against analytic models in the first instance and then inside the RDM against a Monte-Carlo model. This will give us an idea of how accurate each of the models is and how they combine inside the PDE itself. The results will show that whilst some methods may initially appear superior, there is a trade off between a localised accuracy and the question of a consistent global model.

7.2.1 Form Factor Models

In this section we will look at comparing the Form Factor models. We are looking to compare our two approaches to solving for the Form Factor, the point collocation and the Galerkin. It has previously been reported [99] that a point collocation method may produce better results, although this result seems slightly counter-intuitive due to the nature of the approximations involved. To clarify this we will compare point value calculations for each model on the spherical case, then we will compare a patch calculation in a simple parallel square domain to Schroeder's analytic model [84], and

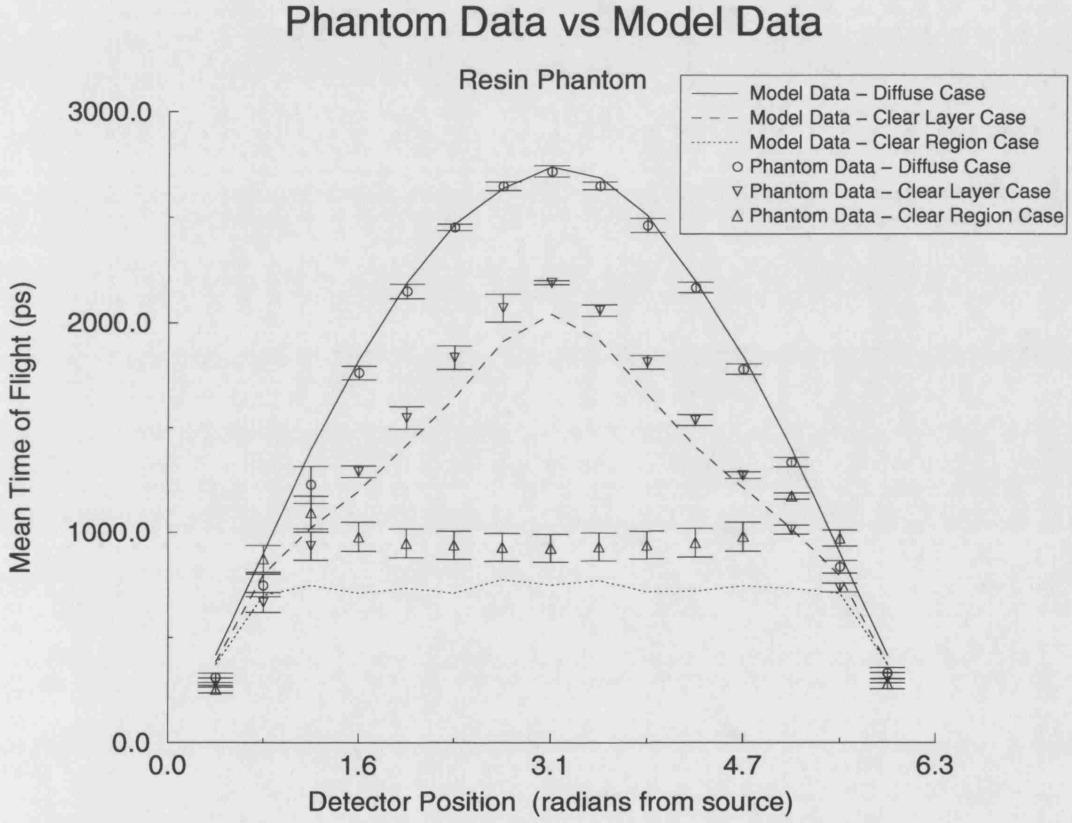


Figure 7.9: A graph showing the comparison of mean-time of flight for the RDM and a glass and resin based phantom.

finally we will compare the two models within a full RDM solution on our standard test case against a Monte-Carlo simulation.

Point-wise Comparison

Here we will compare the point-wise solution of the two Form Factor models to the point-wise analytic solution developed for the concentric sphere case. The idea will be to compare the best and worst case for each model against the analytic model to give and idea of overall performance. The best case in each model is easily seen to be at the point of calculation, where we are setting the error to zero in the model, i.e. at the centre of each side in the point collocation approach and at the nodes in the Galerkin model. The worst case is harder to define, but if we select the points furthest from the point of calculation this is a suitable sample case.

In figure 7.10, we show the best and worst cases for the concentric sphere model in both the outer-to-outer and outer-to-inner case. In each of the graphs we present a line showing the analytic point-wise solution and a point cloud representing the values calculated by our two algorithms over a given spherical mesh. The four graphs represent the following information:

- Figure 7.10(a) shows the outer sphere to inner sphere Form Factor for the worst case.
- Figure 7.10(b) shows the outer sphere to inner sphere Form Factor for the best case.
- Figure 7.10(c) shows the outer sphere to outer sphere Form Factor for the worst case.
- Figure 7.10(d) shows the outer sphere to outer sphere Form Factor for the best case.

Examining these graphs we see that both models handle the calculation well at the point of calculation, which is to be expected. However if we examine the worst case, we see a distinct improvement by the Galerkin over the point collocation. If we examine figure 7.10a, we can interpret this error easily, we see that the error appears like a translation of the curve. This is because we are effectively using the Form Factor from the centre

of the sides at the edges of the sides, meaning we have a translation in radians from the actual calculation.

Patch-Wise Analytic comparison

The next stage is to take a comparison of the patch-wise Form Factor calculation in each model with an analytic model. Here we will use the model presented in [84], which requires completely mutually visible planar polygons. Our chosen test case is illustrated in figure 7.11, as we know that as you reduce the separation to area ratio in an area Form Factor calculation, the error will tend to rise, we will vary h whilst keeping the other components constant. This will allow us to test the robustness of the models as well as their comparative quality. The code used for the analytic Form Factors is provided by the authors of [84] and is down-loadable for academic purposes [83].

There are two important factors to note here, both are consequences of the models. First in the case of the Galerkin model we can easily increase the order of the elements without changing the mesh. This is not so in the point collocation model, here to increase the order of our solution we must re-mesh the problem - as a result of this we show both linear and quadratic results in the Galerkin case. The second point is we have had to rearrange our equations slightly for both models, the point collocation model lends itself to this more directly as we initially calculate over a patch and then integrate with the basis functions of the node - a step we may leave out. In the Galerkin case we must add an additional summation over the patches to get the full values. This slight discrepancy may have some bearing on our results. In table 7.1 we give the numerical comparison of the data as we vary the area to separation ratio, and in figure 7.12 we give a graphical illustration of the results.

We note here that we would expect the Galerkin model to be an improvement even in the linear case. This is simply as it has higher sampling, however we find as was presented in [99] that the point collocation appears better. In the previous case [99] this was arguably as their comparison was unfair - the so-called gold standard used was a different numerical model (the hemi-cube model) which was more similar to the point collocation approach. The implications in our overall model then tend to suggest we would get a lower error and better results from a point collocation approach.

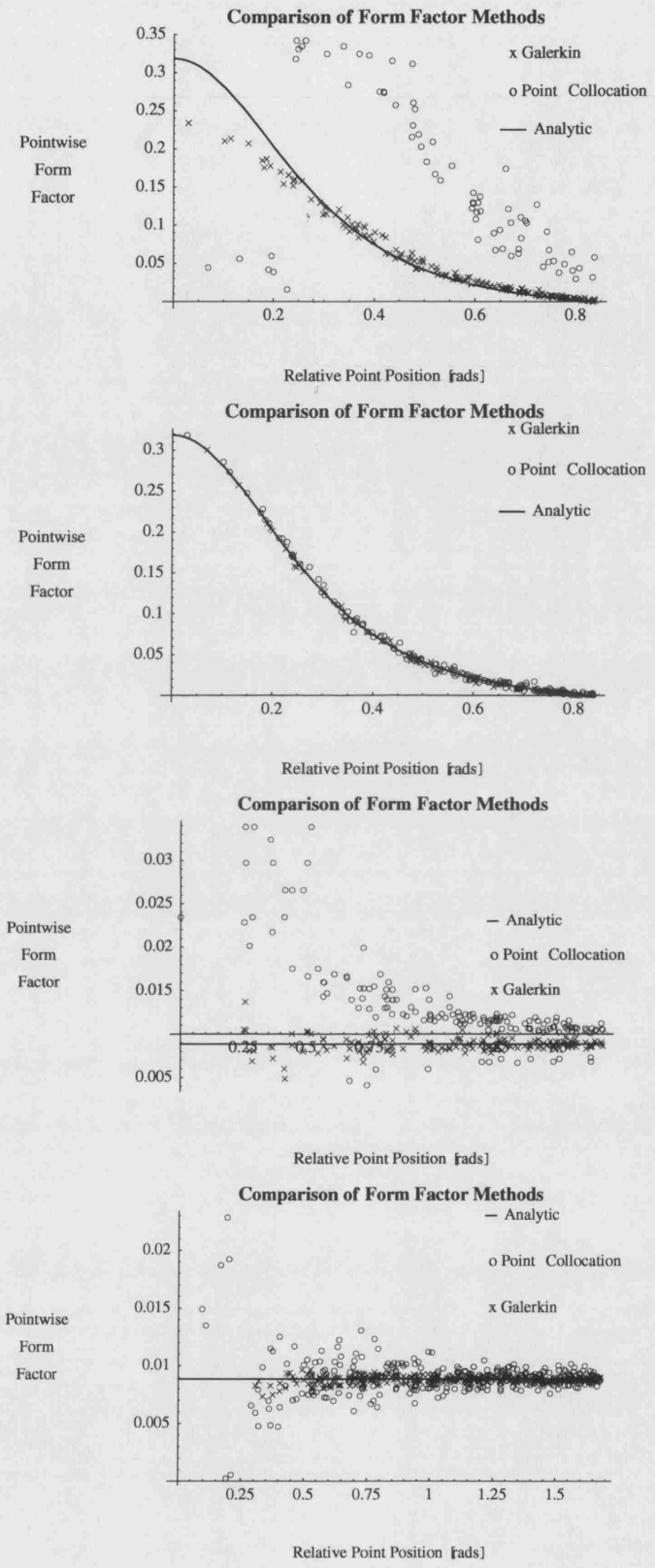


Figure 7.10: Graphs showing the four cases of the error in point-wise Form Factors in our test sphere: (a) outer to inner worst case, (b) outer to inner best case, (c) outer to outer worst case and (d) outer to outer best case.

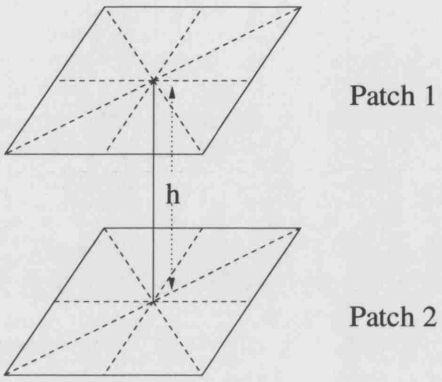


Figure 7.11: A diagram of the geometry for the comparison of the numerical Form Factor methods to an analytic model.

Separation to Size Ratio	1:4	1:2	3:4	1:1
Analytic	1.66101	0.7993	0.442827	0.274358
Point Collocation	1.86815	0.838661	0.456597	0.2800083
Galerkin (linear)	1.46487	0.699422	0.405321	0.258339
Galerkin (quadratic)	1.76345	0.799501	0.442858	0.274368

Table 7.1: A table showing the patch to patch Form Factor results for the case of two parallel squares.

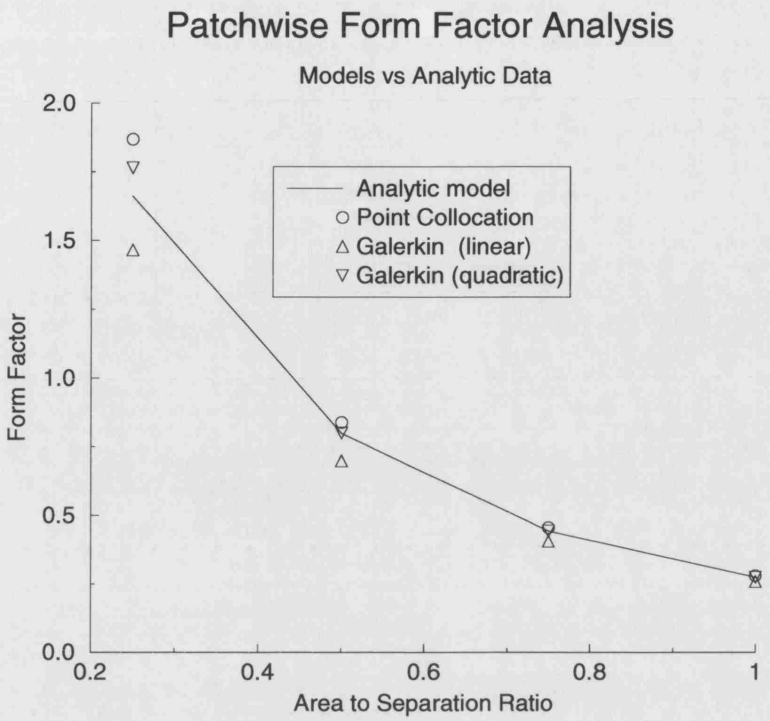


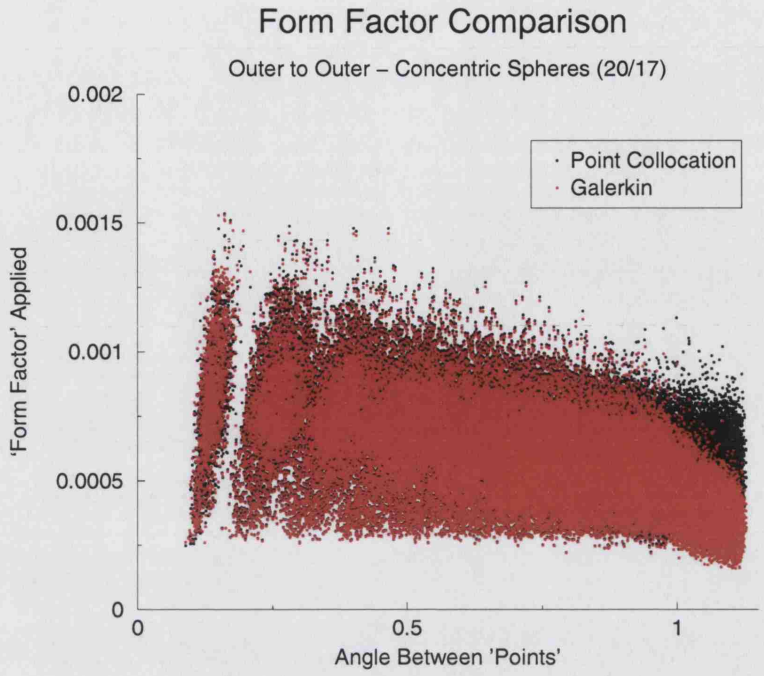
Figure 7.12: A graph showing the comparison of the Form Factor models against an analytic code on a parallel square problem.

Patch-wise Numerical Comparison

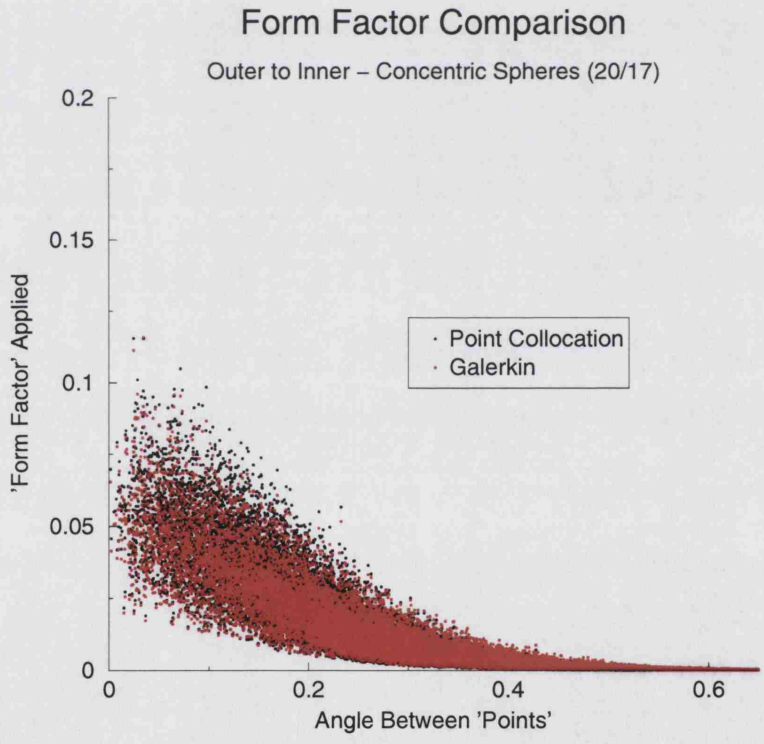
Here we are simply trying to get an idea of the comparison of the models as distributed over a sample mesh. The idea is that as the mesh has a relatively even distribution in element size and shape we should get a fairly compact distribution of Form Factors. To illustrate this we have chosen the case of the outer-to-outer Form Factor on a sphere, as we know the point-wise value is a constant we would predict a flat and tight spread of Form Factor values. This is a sanity check but should also give us more of an idea of robustness of the two models. In figure 7.13 we see that both models are indeed giving a fairly small distribution of values, but that the Galerkin model does seem to show a slightly more compact range. This suggests from our assumptions that it is slightly more robust, but this is purely speculative and we will seek confirmation of this in a full RDM solution comparison in the next section.

Full Model Monte-Carlo Comparison

The final approach to comparing the two Form Factor models is to incorporate them in a full RDM problem and compare them to the data from a Monte-Carlo Simulation.



a)



b)

Figure 7.13: Graphs showing the Form Factors generated by the two different methods for the concentric sphere case, shown against the relative node positions. a) Illustrating the outer surface to outer surface Form Factor and b) illustrating the outer surface to inner surface Form Factor

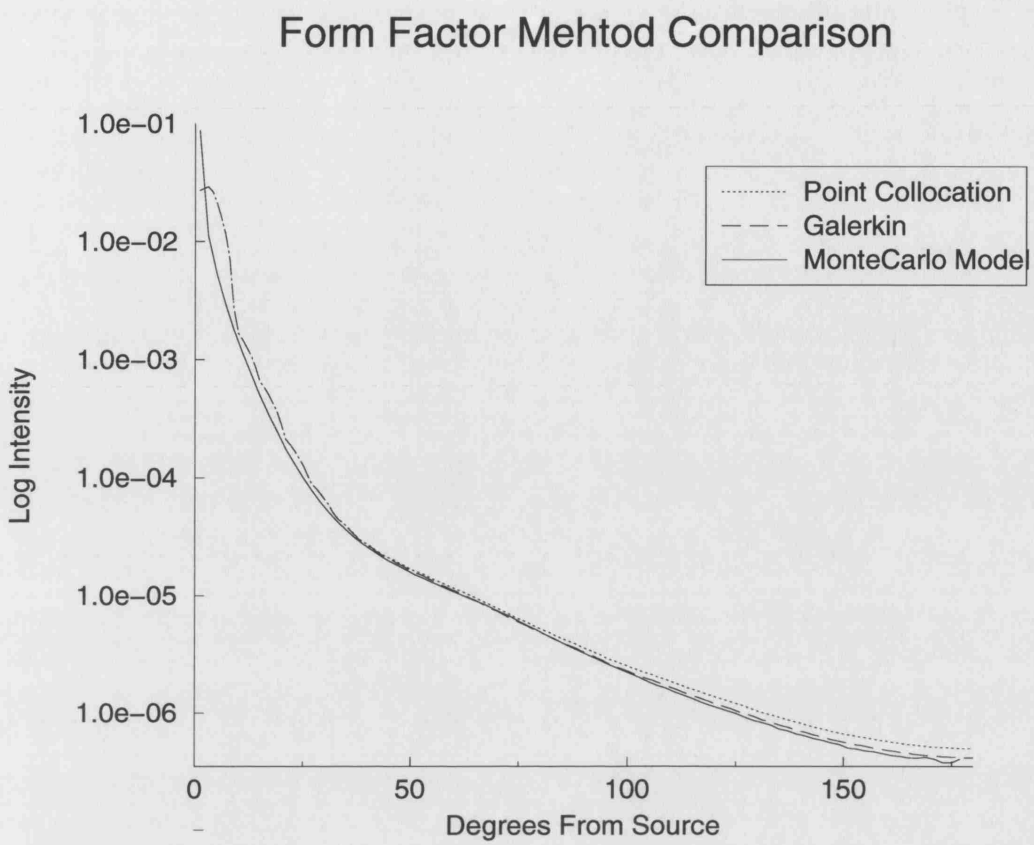


Figure 7.14: A graph showing the comparison of data using the two different Form Factor models, with the simple visibility model in comparison to a Monte-Carlo simulation.

The objective here is to establish which model performs best in the overall simulation of light transport in diffusing domains containing non-scattering spaces.

In figure 7.14 we show that both the Form Factor models produce reasonable data in the RDM compared to a Monte-Carlo simulation of a $25-20-27$mm radius concentric sphere void with the parameters $\mu_a = 0.01\text{mm}^{-1}$ and $\mu_s = 1\text{mm}^{-1}$ with a refractive index of 1.4. If we examine the curves closely however (see figure 7.15) we see that the Galerkin model shows distinct improvement over the point collocation model at this point.

There are several possible reasons for this improvement. First we expect the higher sampling of the Galerkin model to provide a better result, but as we have already seen this may not be the case. The second relates to the visibility issue, as the Galerkin model samples more points we may have the effect that the visibility being better sam-

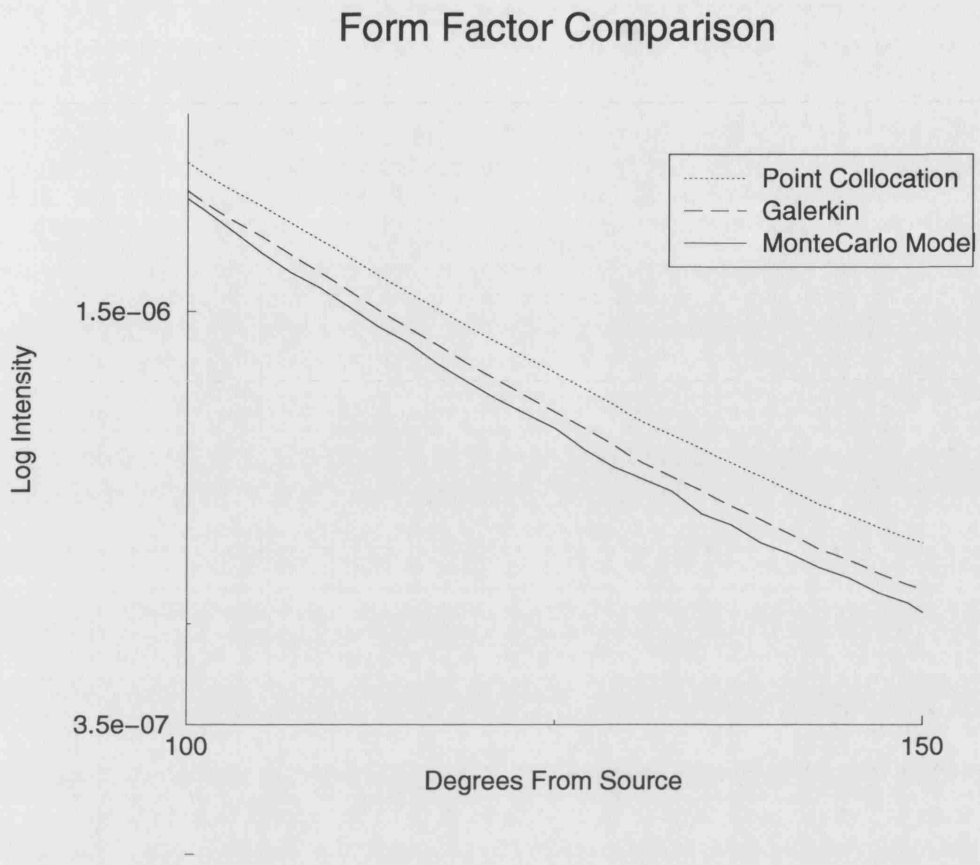


Figure 7.15: A close up of the graph shown in figure 7.14 illustrating more clearly the difference in the models.

pled improves our results, we will see similar effects later from changing our visibility models. Finally another distinct possibility is the choice of error metrics throughout the system, as our Diffusion Approximation is using a Galerkin model it seems likely that matching the error metric throughout the domain will provide a better solution than varying it in the different optical regions.

7.2.2 Visibility Models

Having compared the various Form Factor models, we must now validate and compare the visibility models used in the RDM. The first step is to ensure the models are accurate in comparison to a gold standard, and then we must compare how they operate in the full RDM. We also analyse the errors produced and how they effect the usability of the different models, a factor which was highlighted during some of the experiments conducted in testing.

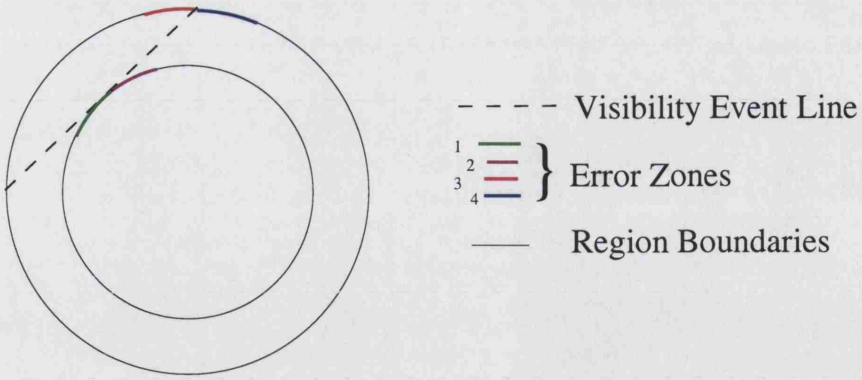


Figure 7.16: A diagram illustrating the four possible zones of error in the visibility algorithms.

Analysing the working of the two visibility models

In order to analyse the errors produced by the visibility models we must first identify the various zones in which the models can fail. That way we can examine why each model is inaccurate. To do this we must first identify our gold standard, as we have an analytic model for a concentric sphere geometry this would appear to be the best gold standard available to us. In figure 7.16 we show the four possible zones where our visibility model maybe inaccurate.

The first stage of comparison is to take a mesh of the concentric sphere geometry (our chosen surfaces are at 20mm and 17mm radii). We may then generate the visibility via both the OpenGL model and the Voxel Model and compare the error in each zone in terms of the maximum angle of error and the percentage of calculations within this angle that are incorrect. This will give us an estimate of how accurate the model is and where its inaccuracies will lie. Knowing where the inaccuracies lie will enable us to analyse the errors and determine the suitability of each model to a given domain.

Table 7.2 gives us a numerical analysis of the errors of the visibility models. It clearly shows that the error in the OpenGL case initially appears to be more random than that in the voxel algorithm, as the voxel errors are confined to specific parts of the possible error zones. If we examine these errors we can actually interpret them in terms of the algorithms which leads us to some conclusions about their relative usefulness. We note at this time that the worst angular error here is equivalent to less than a millimetre in geometric difference - the significance of this will become clear in section

Zone	OpenGL		Voxel	
	Maximum Error Angle	Percentage Error	Maximum Error Angle	Percentage Error
1	3.2506	35	0	0
2	0.155066	$\ll 1$	3.72461	62
3	0.216112	33	0	0
4	0.176489	5	0.278179	40

Table 7.2: A table showing the scale and location of the errors for the two visibility models.

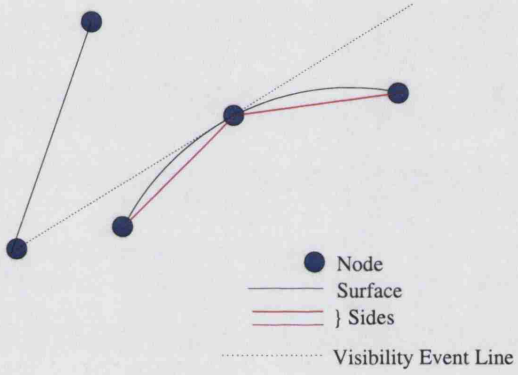


Figure 7.17: A diagram illustrating why region 1 is the most likely place of error in our OpenGL algorithm.

7.6.3.

If we consider the OpenGL algorithm and look at a node at the edge of the visible space we can see where errors will begin to develop. Looking at figure 7.17 we can see how the errors in the OpenGL algorithm begin to appear, nodes at a visibility event or close to it will have sides which either are not visible when they should be or vice versa.

This leads us to an unfortunate conclusion about the OpenGL visibility model, that it is only really sufficient for smooth surfaces. We can see this more clearly if we look at figure 7.18, where we have depicted a rough surface mesh. In this case the OpenGL algorithm will predict no visible nodes on the drawn surface leading to no light coupling, this is obviously a completely inaccurate result. However given a

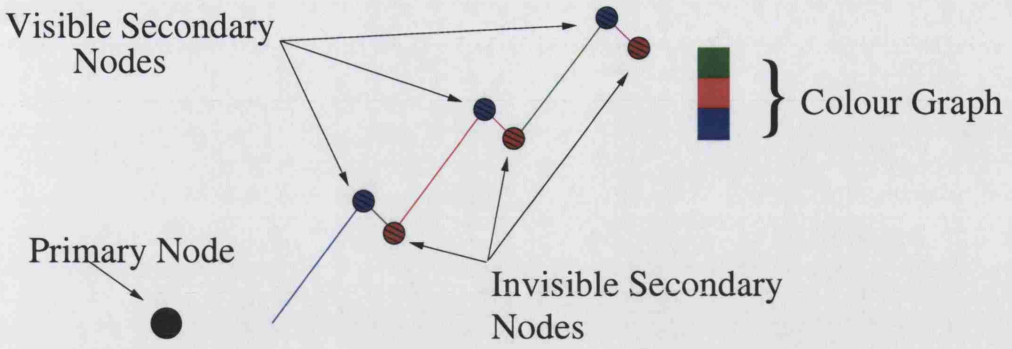


Figure 7.18: A diagram illustrating the problem of visibility from colour-coding the sides in a rough environment, the colour-graph of the primary node clearly shows none of the secondary nodes are visible - an error

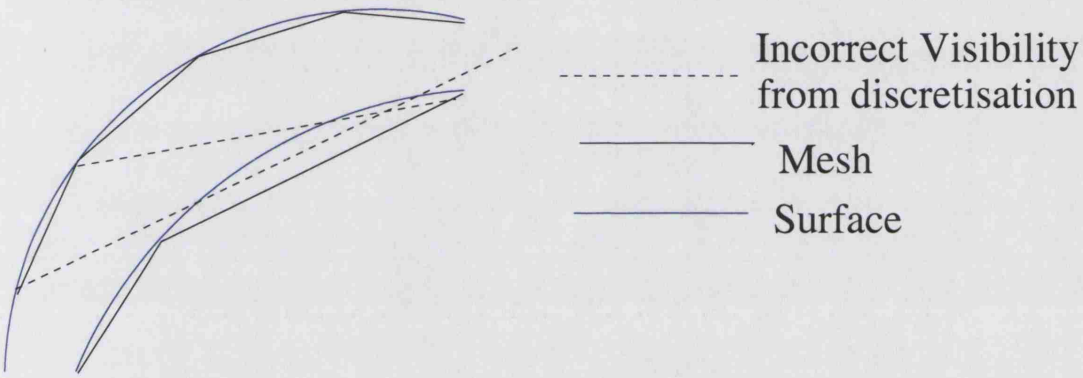


Figure 7.19: A diagram illustrating how the errors arise in the voxel visibility not from the visibility model, but from the surface discretisation.

smooth surface such as a sphere, from the initial estimate of error we would predict a reasonable result, this is confirmed below in the subsection on evaluating the use of the visibility models within the RDM.

Unlike the OpenGL error, the voxel visibility error appears systematic, if we consider figure 7.19 we can see why. In this figure we illustrate the fact that actually we are not dealing with an error in the algorithm, but in reality it is within our interpretation of the gold standard. The visibility generated by the voxel algorithm is actually correct for the geometric approximation we have of the concentric spheres, whereas the gold standard is correct for the true geometry.

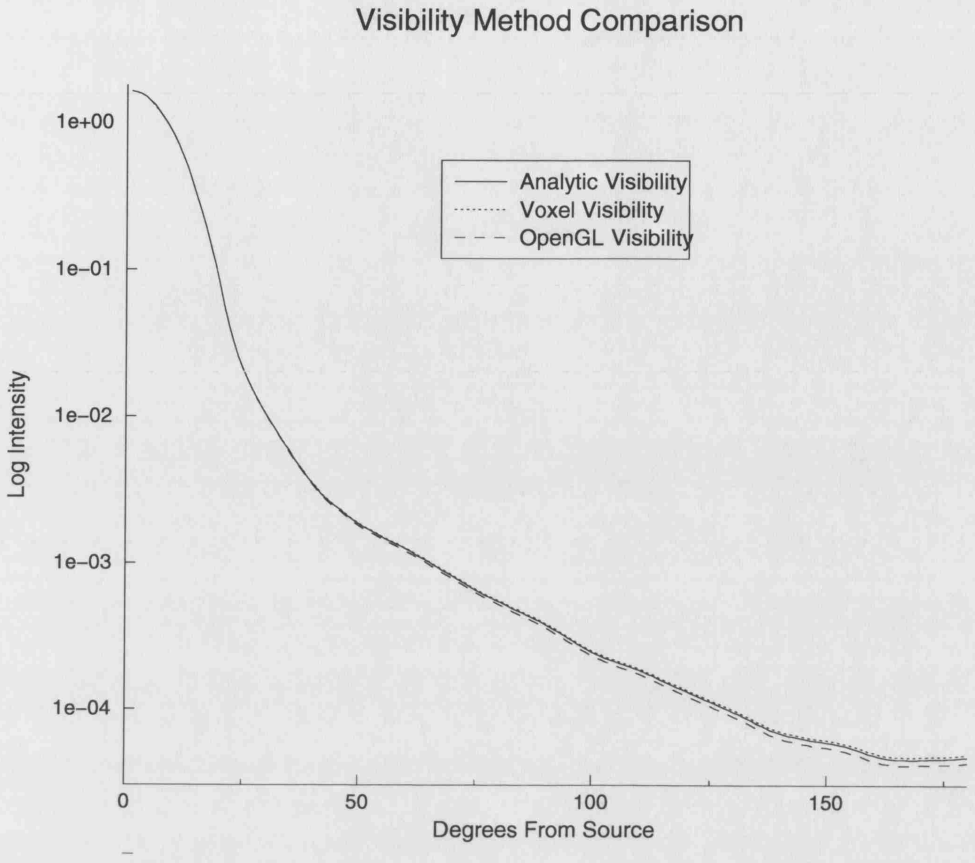


Figure 7.20: A graph showing the comparison of data using the two different generic visibility models compared to using the analytic model in the spherical case.

Evaluating the models in the RDM

Having established that both the voxel and OpenGL models are reasonable visibility models for a smooth concentric spherical domain, we move on to examine whether they are sufficient to handle the problem posed in the RDM. To do this we compare them in our simple spherical model case to a solution generated using the analytic visibility model.

Examining the boundary intensity as shown in figure 7.20 we see the three case produce almost identical data. It is important to note that in the tail of the data, far from the source we can see some variations between the three models, see figure 7.21. These variations are consistent with the error regions we obtained previously with the voxel model piping more light around the void and the OpenGL model retarding the light flow around the void. The significance of these errors will become clear in section

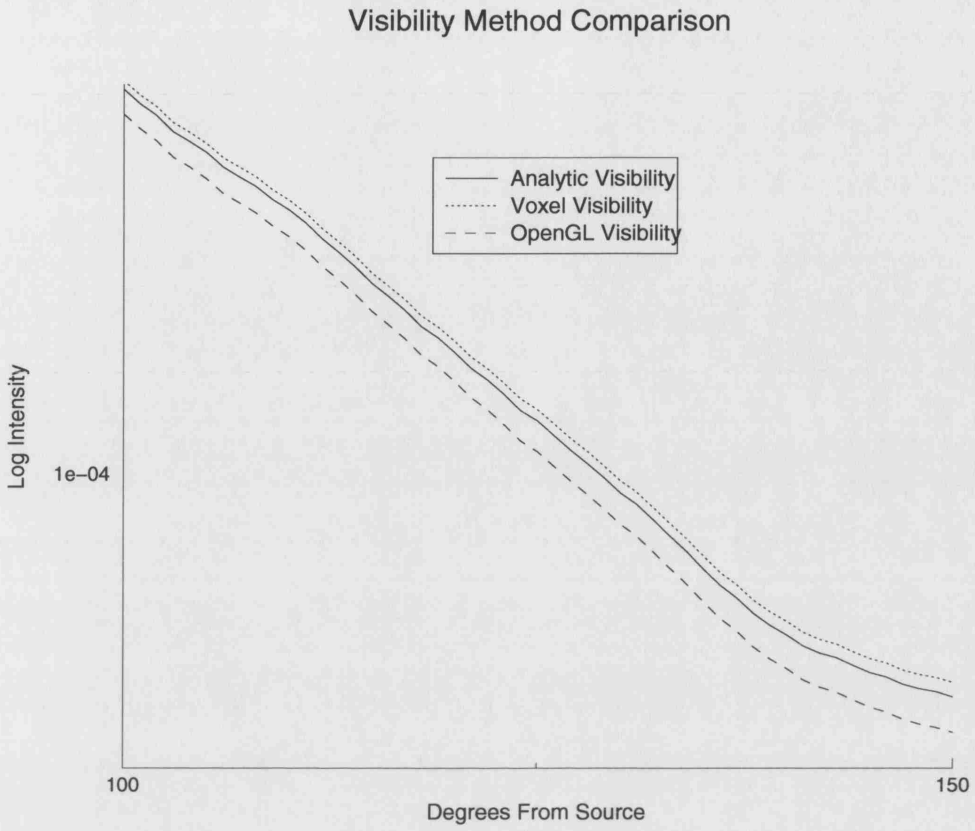


Figure 7.21: A close up of graph 7.20 showing how the two different generic models compare to the analytic case.

7.6.3. However at this time we appear to have two visibility models which function for our problem, we note that in more complex geometries, it will be better to rely on a voxel model as its errors stem purely from our geometric representation rather than a more subtle question of whether the domain is within the scope of the model as is the case for the OpenGL model.

7.2.3 Visibility Representations

The final thing we must consider is how effective the process of shadow masking is in improving our solution. In order to do this we compare the four methods described in chapter 6, these are marked in the graphs as follows:

- Method 1a - Point collocation Form Factors with global visibility
- Method 1b - Point collocation Form Factors with shadow masking
- Method 2a - Galerkin Form Factors with global visibility

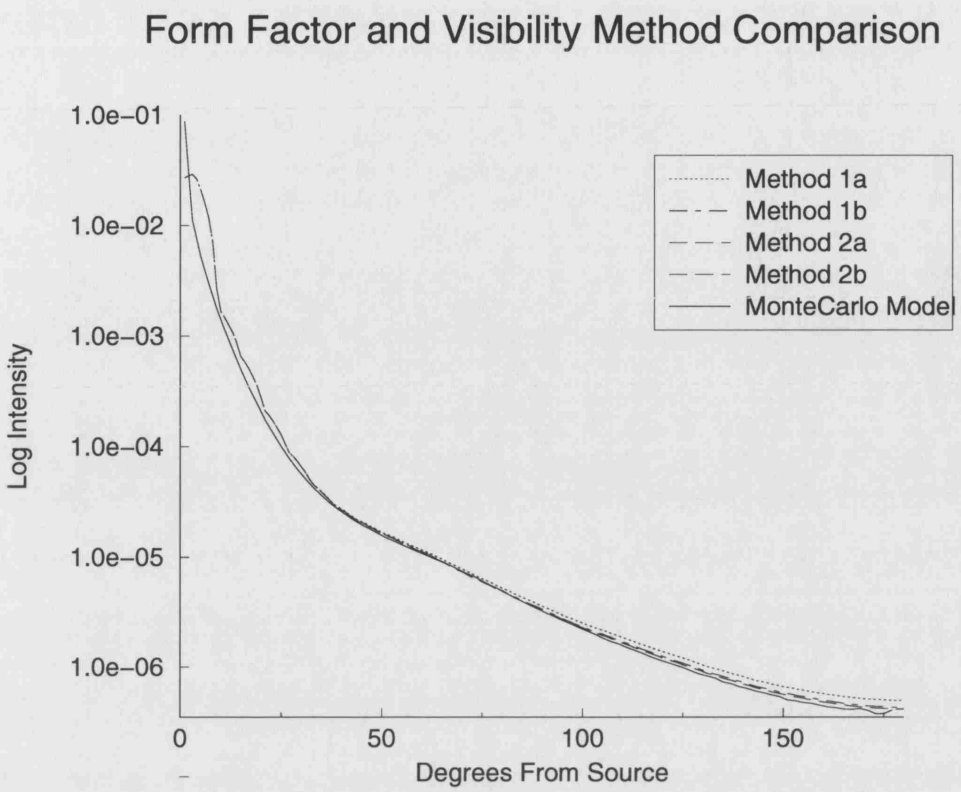


Figure 7.22: A graph showing the comparison of data using the two different Form Factor models, and the effect on them of the different visibility representations.

- Method 2b - Galerkin Form Factors with shadow masking

As a reminder global visibility is where we consider the visibility only at the nodes in question, whereas the shadow masking approach compares each component of the Form Factor summation and checks its visibility to test for inclusion in the final value.

On examining the graph in figure 7.22 it is unclear how much improvement we have although all models provide a close match to the Monte-Carlo simulation. If we examine the close-up provided in figure 7.23 we can clearly see that in the case of the point collocation method we get clear improvement on using the shadow masking algorithm although it still fails to match the Galerkin approach. In the case of the Galerkin however we see little or no change to the model, there are two factors here, first the model may be reaching an optimum for the approximation, second at this time we have not introduced a visibility fringe. In the case of the point collocation model we may use the same visibility structure to construct our mesh under shadow masking,

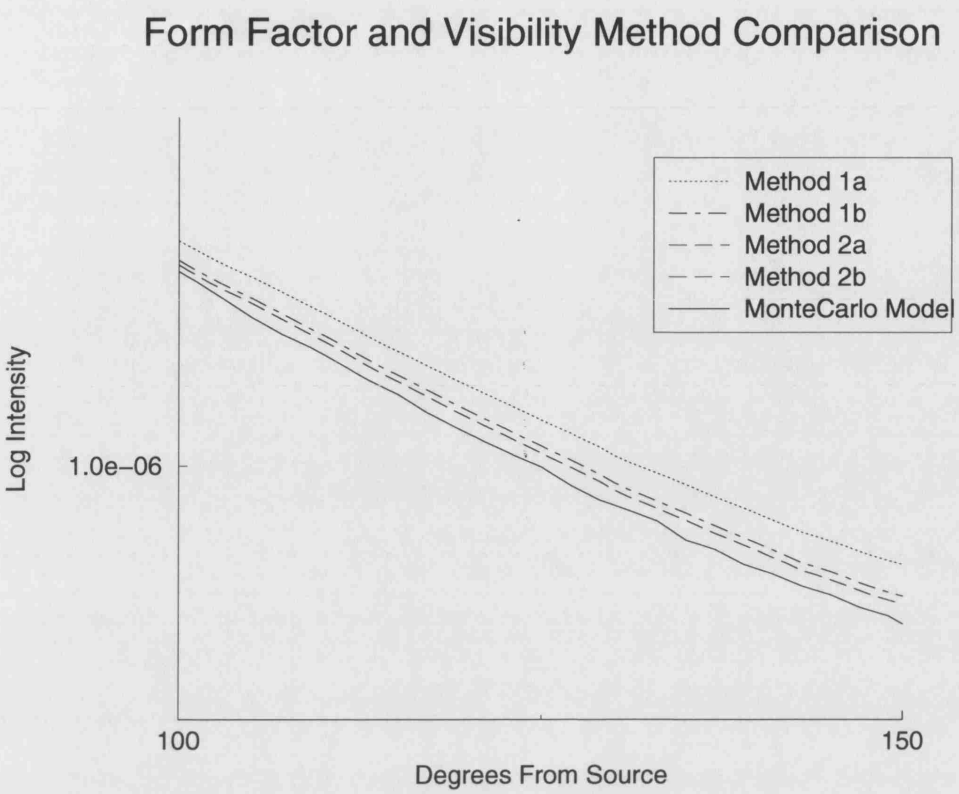


Figure 7.23: A close up of the graph in 7.22 illustrating how the visibility representation improves on method 1, but has little effect on method 2

however if we use the Galerkin approach we need to add extra slots in our matrix for entries where two nodes are not visible to each other but two of their neighbours are. The contribution here is considered small compared to the extra computational burden imposed by such an approach. It may be worth examining at a later time, but given results shown later we consider any of the current models sufficient for what we are able to achieve in reasonable computational bounds with this model. This is one possible extension which we leave as a possibility for future work when computational bounds increase.

7.2.4 The chosen model for further analysis

At this point we will mention that for now with all experiments we will be using the shadow masking Galerkin Form Factor model. The reasons for this are two-fold, first it is the best method of our selection in matching the Monte-Carlo simulation with the RDM and second, it is actually the cheapest computationally in terms of calculation time.

7.3 Model Comparisons II - Localised P-Refinement and Simple Meshless models

In this section we illustrate the working of the geometric models developed to improve the computational costs of the RDM. The first is the implementation of localised p-refinement to reduce the mesh size and hence memory costs of the problem. The second is the novel simple meshless model designed to eliminate the need for complex geometric meshing in irregular domains. The latter will provide us a tool useful for further examination of the model in terms of physical validation, and also for looking at how stable the model is in terms of a gradually varying geometry, these results will be presented in the next section (section 7.4).

7.3.1 Localised P-Refinement

The first of the geometric models we developed to reduce the computational expense of our problem was to implement a localised p-refinement strategy. In this section we will first illustrate it working in a 2D diffusive environment (used to develop the model for its simpler implementation). We go on to examine its effects and usefulness in a 3D model, we start by demonstrating its usefulness in a simple diffusion based model, and then finally we will examine whether we can make use of it to reduce the cost of the 3D RDM problem.

2D - Diffusion based

In this section we start with a simple illustration of how using a localised p-refinement model reduces our computational burden. We take a linear triangular mesh (see figure 7.24), representing a diffusive domain, and show the fields under a selection of localised p-refinements. The mesh used is extremely coarse and is designed to allow a quick qualitative demonstration of the meshing problem and how p-refinement helps in particular how a selective p-refinement can reduce computational cost.

In figure 7.25 we illustrate the basic effect of a global increase in the p-refinement of the mesh. It shows the effort of using the linear, quadratic and cubic versions of the given mesh[1]

[1]The field is not shown logarithmically for the linear case as it contains negative values, a sure sign that this discretisation is not correctly representing the physics of the problem.

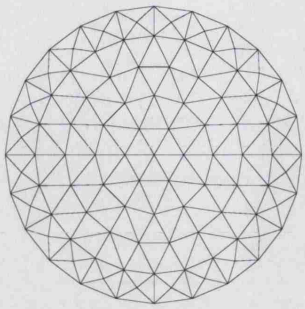


Figure 7.24: A diagram showing the element arrangement in the mesh used for the 2D p-refinement tests.

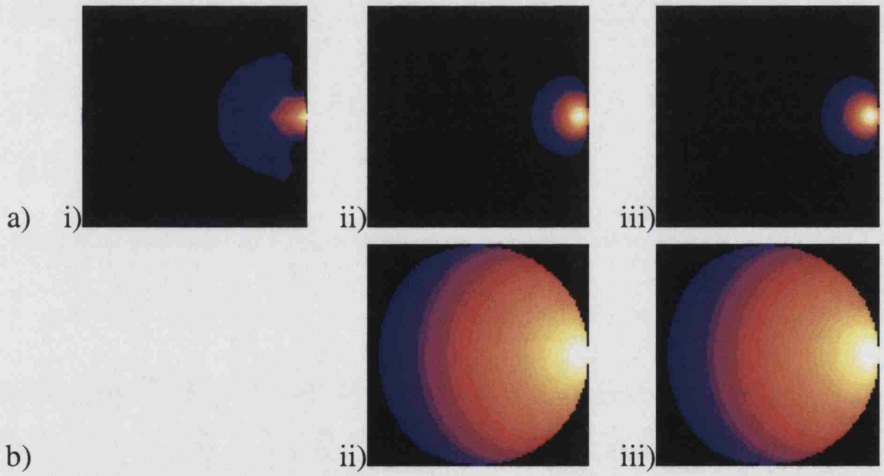


Figure 7.25: Illustrating the effect of p-refinement on the field. Images a) i-iii) show the linear, quadratic and cubic fields for the case for a point isotropic source $\frac{1}{\mu_s}$ below the surface. Images b) ii) and iii) show the logarithmic plot of these fields for the quadratic/cubic case.

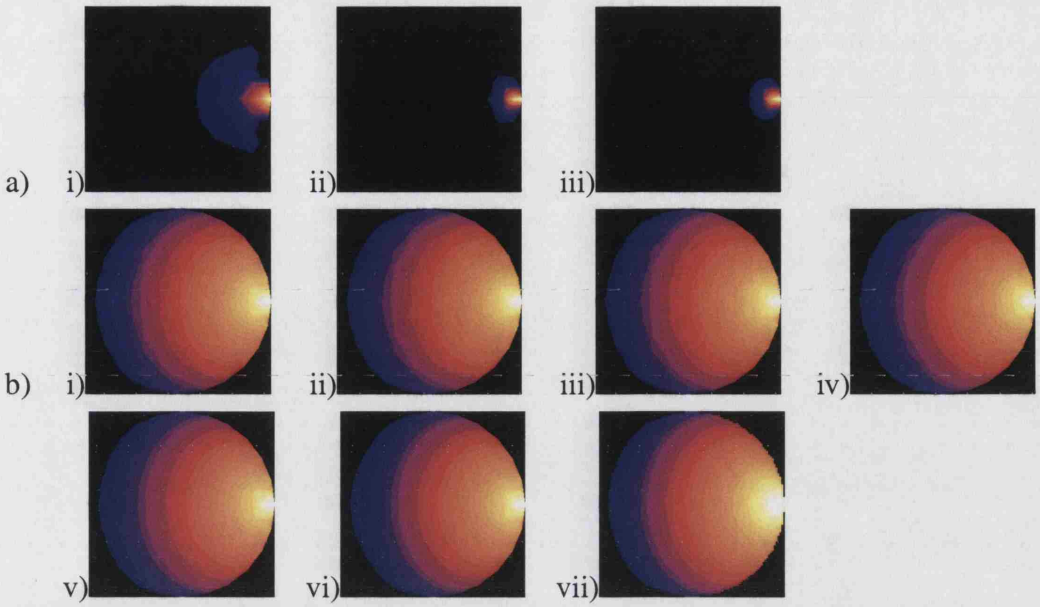


Figure 7.26: Illustrating the effect of p-refinement on the field. Fields generated with an increasing proportion of quadratic elements - a circle of increasing radius is used (a step of 7mm at each stage), centred at the edge nearest the source, to identify the quadratic zone. Set (a) is a linear set up to quadratic radius 14 (0,7,14),whilst set (b) is logarithmic plots from 14 to 56 (14,21,28,35,42,49,56) with 56 being fully quadratic.

It is clear here that the linear model is insufficient at this level of h-refinement, but that both the quadratic and cubic fields appear similar. In an effort to demonstrate the usefulness of the graded p-refinement we show the fields for a simple regime of increasing the level of p-refinement in the mesh from linear to quadratic over a gradually increasing sub-domain. Figure 7.26 shows the effect on the field of gradually increasing the amount of quadratic refinement in the mesh. It is done by converting all elements wholly contained in a circle centred at the edge of the mesh near the source to quadratic and using a layer of linear-quadratic elements to connect these to the rest of the linear mesh. The radius of the quadratic region is gradually increased, the resulting images are shown logarithmically where the field was entirely positive and linearly otherwise.

These results give us a clear indication that we can use a localised p-refinement process to reduce the number of nodes needed to represent the field. We can clearly see that whilst we cannot simulate the diffusion problem on this level of geometric meshing

with linear elements, we can with quadratic ones. However it is not necessary to use quadratic elements throughout the domain. We only need to use them in a limited space to ensure we have a reasonable simulation of light transport for this case. In other words we should be able to use this approach to reduce the number of nodes and therefore the computational costs of simulation. In the following section we will go on to illustrate how these effects are carried through into the 3D problem.

3D

In this section we will show two cases for the field generated in the presence of linear and quadratic elements. One a homogeneous solid brick, and the second our standard concentric sphere void model. We illustrate the effect of gradually increasing the quadratic component radially from the sources/detectors into a linear mesh. This is done as if we use a radial function then we can ensure that only the limited number of elements defined in section 6.3 are required to generate the mesh. Given this radial model the element can be identified as either edge, side or fully quadratic. Another approach shown here is the use of quadratic elements where the field becomes unstable (or negative) in the case of the solid slab this is much more effective, but in the void model this approach does not yield productive (computationally saving) results.

Solid Slab - Diffusion Based

In this section we examine the effect of the use of p-refinement in a slab geometry (40mm x 40mm x 70mm). The sources are arranged on one face and the detectors on the opposite face, as shown in figure 7.27. The parameters of the domain are our standard, $\mu_a = 0.01\text{mm}^{-1}$ and $\mu_s = 1.0\text{mm}^{-1}$, with a refractive index set at 1.4.

In our chosen tetrahedral sampling we have arranged that the linear model is insufficient to provide physically real data. In other words we have a system which produces negative photon density. We have then used two strategies to increase the sampling to provide a correct model. The approaches are to create regions of quadratic elements either centred at the sources, or in an adaptive way at the negative field points. In table 7.3 we illustrate the results of these two models. Here we are trying to get a quantitative analysis of the improvement and how to use perform the re-sampling strategy. We can see that if we use our original source centred paradigm, even with 89% of the mesh being quadratic we still have a clearly physically incorrect model. However if we use

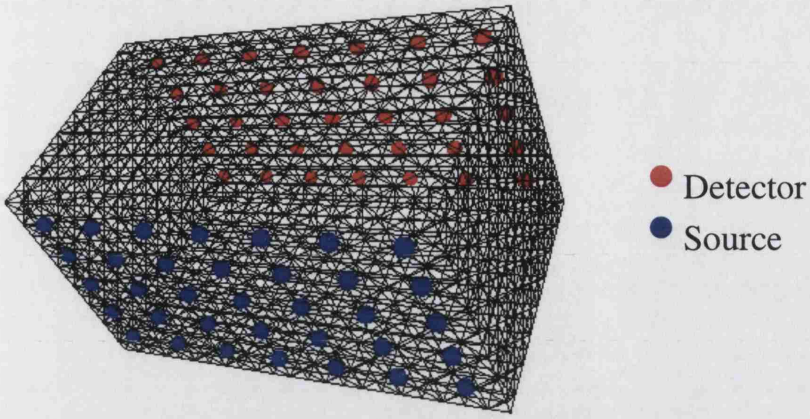


Figure 7.27: Illustrating the source detector geometry for the 3D Localised p-refinement diffusing slab test.

our new error based paradigm, we see that the model becomes realistic with around 36% of the mesh being quadratic and once we reach 60% quadratic elements we are changing the solution by less than 1%, so we are close to a correct model. We note that, there may be a more sophisticated paradigm based on combining these two approaches, but at this stage we move on to examine the effects in the case of primary interest, the RDM.

Concentric Spheres - RDM

In this section we present the initial results for the use of localised p-refinement elements in a void based model. To illustrate the model we start from a coarse mesh, which if fully subdivided produces a working linear mesh (with an equivalent solution being provided if converted to quadratic elements). The idea being to see how many of the extra nodes generated we need to acquire a good result. Here we have grown the p-refinement zone out from the void layer and the boundaries to see how much of the mesh needed to be refined, the idea here being we know that the void being a geometric problem will require a high density of meshing.

In figure 7.28 we illustrate the gradual convergence of the boundary field solution as we increase the proportion of the domain modelled with quadratic elements as opposed to linear. We clearly see an initial jump in the convergence of the solution, this is given at the point where the solution actually becomes physically real - that is

Quadratic Radius	Negative Centred		Source Centred	
	%age quadratic	Minimum field value	%age quadratic	Minimum field value
1	0	-0.00436253	0	-0.00245869
2	0	-0.00436253	0	-0.00246465
3	1.11429	-2.12198×10^{-7}	0.483333	-0.00242701
4	1.41071	-1.21911×10^{-7}	2.96667	-0.00276143
5	4.4119	-1.8132×10^{-10}	6.475	-0.00267982
6	5.20357	-1.84894×10^{-10}	13.7583	-0.00150033
7	8.74048	-7.78072×10^{-11}	21.7417	-0.00151457
8	11.0512	-1.07406×10^{-10}	27.5833	-0.00160465
9	16.85	-9.35092×10^{-12}	34.775	-0.00191033
10	17.4905	-2.77345×10^{-11}	39.4	-0.00194448
11	24.9012	6.45479×10^{-11}	45.8083	-0.00160907
12	28.0452	4.47074×10^{-11}	49.8	-0.00207759
13	34.844	1.42031×10^{-10}	56.3083	-0.00130735
14	36.9095	1.43165×10^{-10}	60	-0.000943929
15	44.231	1.44762×10^{-10}	67.8	-0.000639493
16	47.7643	1.46345×10^{-10}	70	-0.000459024
17	54.2381	1.47148×10^{-10}	78.4333	-0.000306989
18	59.0286	1.49072×10^{-10}	80	-0.000228515
19	67.0321	1.50743×10^{-10}	89.4	-0.000199483

Table 7.3: A table illustrating the different approaches to using higher order elements to stabilise the system of equations.

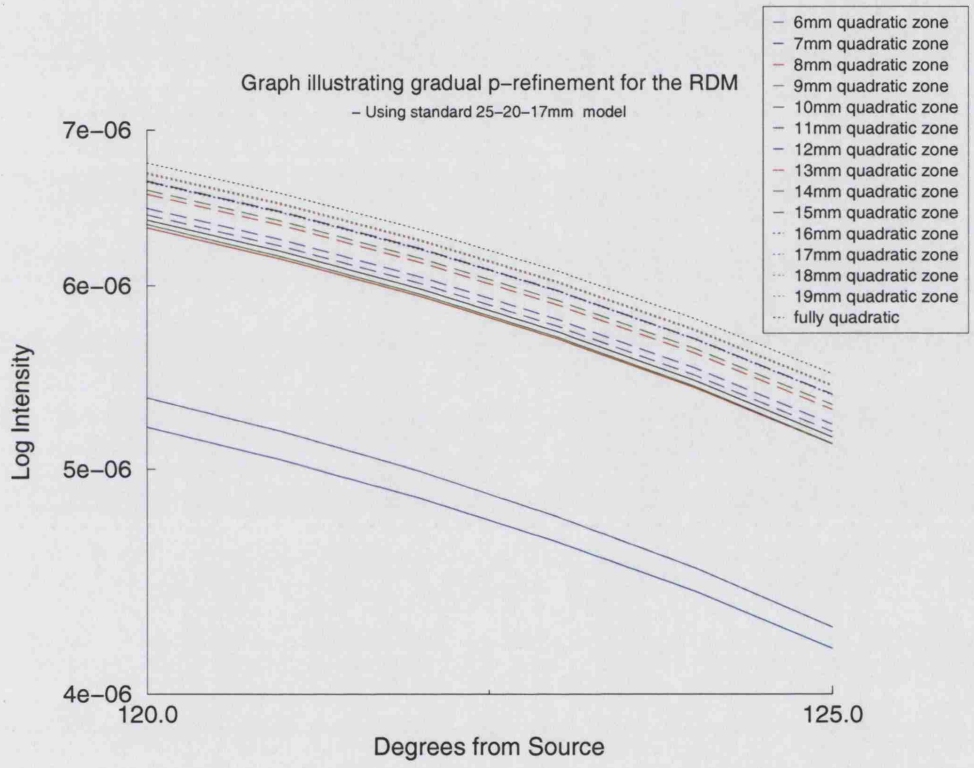


Figure 7.28: A graph illustrating the gradual solution convergence as we increase the quadratic region of a locally p-refined void mesh.

we have a fully positive field throughout the domain. After this we gradually move towards the final solution. What will become clear here is that we actually need a very high density of quadratic region in order to reach the full solution. We note that at this time, the level of meshing needed does not significantly reduce our computational cost in this case. Other similar results have been seen in the other geometries modelled in this thesis. It seems that whilst this approach may be useful in some situations it is not sufficient to the problem we are dealing with in terms of computational cost reduction.

7.3.2 Simple Meshless Method

In section 5.4 we outlined the development of a new meshless FEM model for use in solving the DA/RDM. It relies on an assumption that we can model the boundary of a domain in a separate manner to the domain, provided the our domain and boundary approximations are similar. In this work we are only interested in whether this assumption is sufficient to the problem at hand, so we will not be analysing the precise nature of the limits of the approximation required for the assumption to work. Here we simply look at whether it will work in our given problem type, first in a purely diffusive environment and second in a diffusive environment with non-scattering spaces (our domain of interest). We will then go on to look at internal field and boundary field comparisons in the RDM, showing that the correct simplified meshless model assumption is sufficient for our problem domain.

First we test the model using a typical voxel mesh for handling a spherical geometry. In this case we compare boundary data from a standard voxel model, two types of simple meshless voxel model, tetrahedral model, Monte Carlo Simulation and an analytic code. This should give us a basic validation of our assumption for the types of domain we are using, curved domains with say a 1mm voxel and tissue type optical parameters. We will go on to show how the model performs in the clear region case. Showing examples of internal fields from both tetrahedral models which we know to work and from simple meshless voxel models. These will show us that we have a qualitative comparison and then we will also show a good quantitative comparison of the boundary data indicating that, in this case, our simplifying assumption at the boundary is sufficient to model our domain.

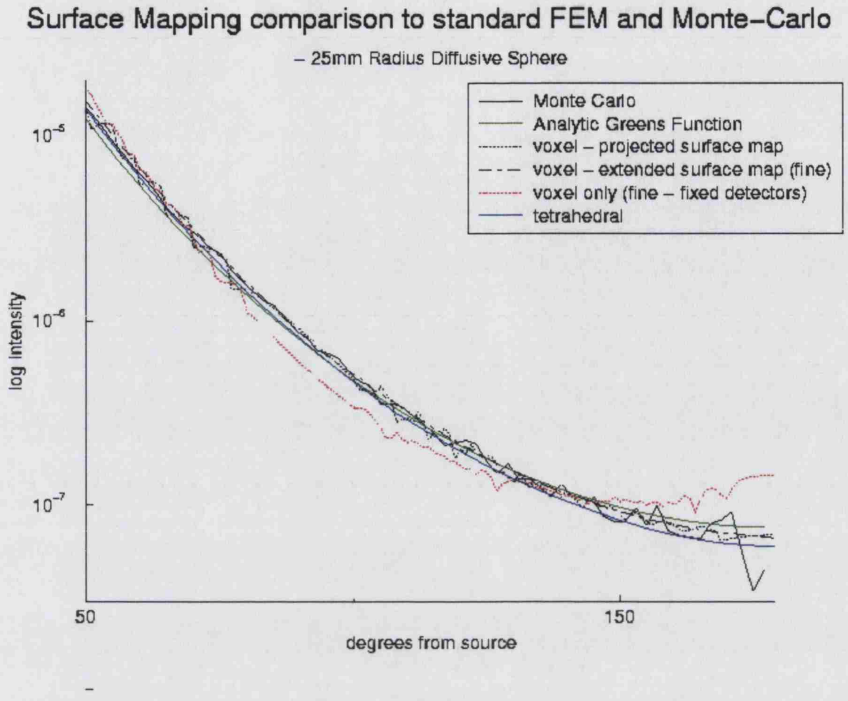


Figure 7.29: A graph, showing the boundary profile of the various FEM models in comparison to Monte-Carlo simulation and the analytic solution for light propagation on a diffusing 25mm radius sphere.

Model validation with the DA

In this section we perform a comparison of models on a 25mm radius diffusing sphere, where $\mu_s = 1\text{mm}^{-1}$, $\mu_a = 0.01\text{mm}^{-1}$, refractive index $= 1.4$ and for Monte Carlo we set $g = 0$. We place a source at $\frac{1}{\mu_s}$mm below the surface and take as our measure the boundary data. We show two models which we consider as our standard to match, the first is a Monte-Carlo simulation of the problem, the second is an analytic model of the DA with Robin boundary conditions for the sphere (provided by S. Arridge).

Figure 7.29 shows our standard models and compares them to four cases, a standard tetrahedral FEM model of the problem, and three voxel cases. We clearly see that in the case of a projected surface voxel mapping code and in the standard voxel model, we can see the shape of the domain - i.e. the voxel edges. However when we use an extended voxel model we get a smooth fitting curve, with the voxel edges disappearing from the data. We also note that in this case the extended voxel model provides a bet-

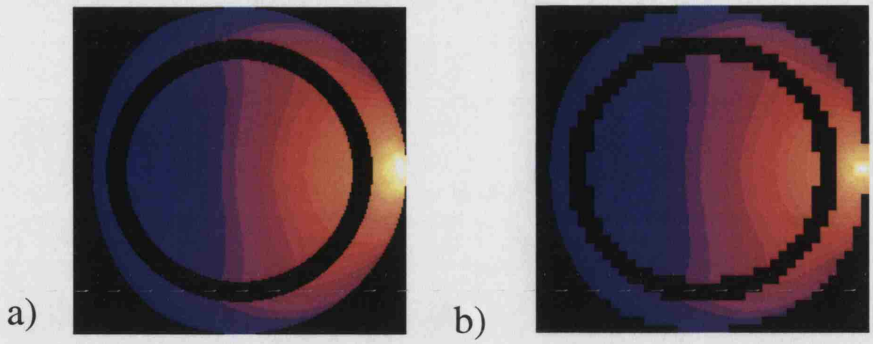


Figure 7.30: A diagram showing the qualitative comparison between (a) an unstructured grid and (b) a surface-mapped voxel model.

ter fit to the Monte-Carlo and analytic models than the unstructured tetrahedral model. Having shown that our assumption is sufficient, at least in the diffusive case, we go on to examine the effects of our new approach on the RDM.

Qualitative Comparison for the RDM

In this section we show a simple qualitative validation of the model. The objective here is as an initial validation that the meshless model provides reasonable internal fields. Here we use our standard test case of a 25mm radius sphere containing a concentric clear layer between 20mm and 17mm radii. The optical properties of the model are $\mu_a = 0.01\text{mm}^{-1}$ everywhere and $\mu_s = 1.0\text{mm}^{-1}$ in the body of the sphere, with $\mu_s = 0.0$ in the clear layer.

In figure 7.30 we clearly see that the voxel model is capable of representing the void model in a simple qualitative comparison, we note that attempting to run a void model on a standard voxel code fails if we use a mesh-correct visibility model and Form Factor calculation - i.e. we cannot solve the system matrix. In the next section we will continue by demonstrating that the simple meshless method produces good quantitative results as well.

Quantitative Comparison for the RDM

In this section we will illustrate a quantitative comparison between the unstructured RDM and two mapped surface RDM models as well as to a Monte-Carlo Simulation. The models are as in the qualitative example given above, but for the Monte-Carlo model we use $\mu_s = 1 \times 10^{-7}\text{mm}^{-1}$ in the clear layer, note we are using a refractive

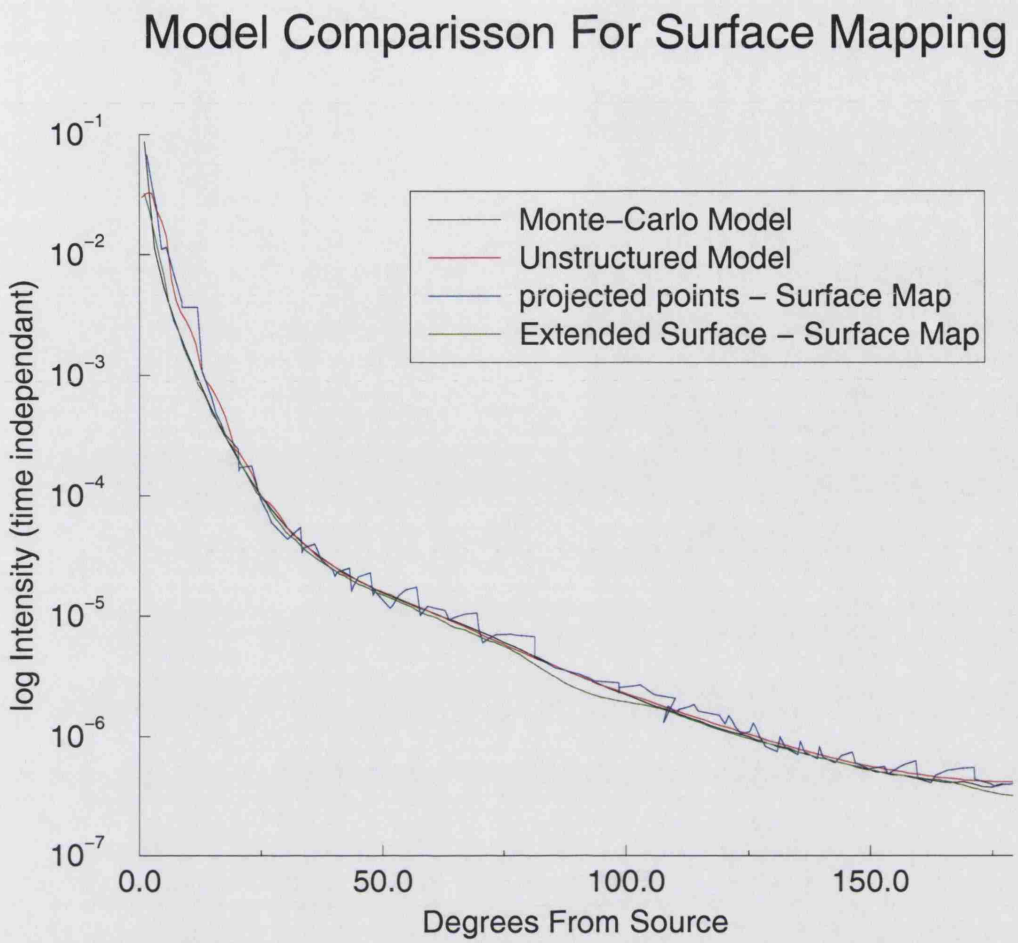


Figure 7.31: A diagram showing a quantitative comparison between the structured RDM, Monte-Carlo simulation and two surface mapping models.

index of 1.4 and g is set to 0 as before. The two surface map models are distinguished as follows, the projected points model where at the outer boundary we have used a voxel selection such that voxel τ is included if more than half of it is contained in the domain Ω, and the extended voxel model where a voxel is included if any part of it is included in Ω. From figure 7.31 we can clearly see that in both cases we have a good simulation, but that for the projected points model we can see the data is structured by the voxel surface. This indicates that our assumption regarding the use of a corrected boundary in place of the mesh boundary is sufficient in our case, using 1mm voxels.

7.4 Model Validation Revisited

Now that we have a working voxel model it is easier to create meshes of general geometries, it is also possible to create computationally feasible meshes for some larger

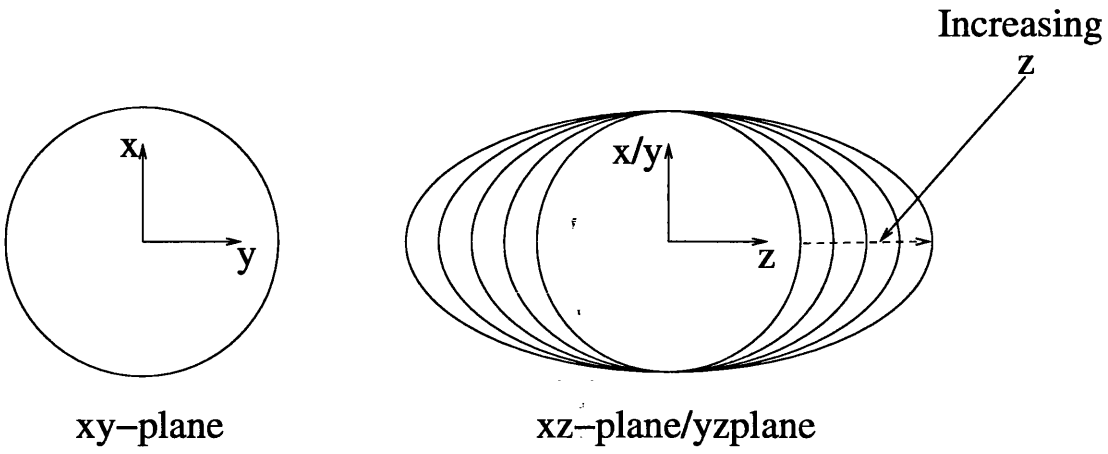


Figure 7.32: A diagram showing the x-y geometry of our ellipse model and the varying geometry in the x-z/y-z plane

geometries. In this section we present two results illustrating further validation of the model. First by using a gradually changing geometry we illustrate that our forward model is stable, demonstrating a smooth change in data for a smooth change in geometry. The second result is more significant, we illustrate a refractive index matched physical phantom and show a good correlation with the RDM. This allows us to clearly demonstrate the correct physical functioning of the RDM.

7.4.1 A stable model

In this section we will demonstrate that the RDM is a stable forward model, by using a layered elliptical geometry with a varying z-axis. We start from our classic $25 - 20 -$ 17mm sphere model and gradually stretch the z-axis, (as illustrated in figure 7.32). The void remains at a depth of 5mm and with a width of 3mm - i.e. at a z-axis of 40mm on the outer shell we have an outer void surface at 35mm and an inner void surface of 32mm in the z axis.

In figure 7.33 we show the boundary data for the x-y plane for a range of z-axes from 25mm to 45mm we clearly see here a gradual change in the data as the axis increases. This indicates a stable forward model that shows the expected gradual change in the data as we gradually vary the domain. This is expected but important as it shows we do have a system stable to change and therefore reliable for the purposes of the inverse problem.

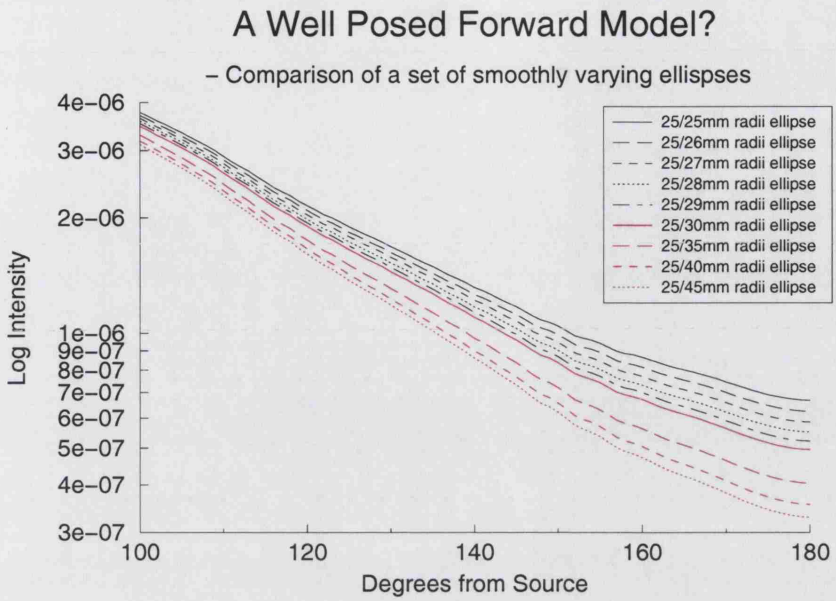


Figure 7.33: A graph illustrating how as we gradually increase the size of our ellipse the field pattern gradually changes.

7.4.2 Physical Validation - a completely Resin-based Experiment

The other important validation task we can now undertake is a concentric cylindrical geometry on the scale of a phantom for MONSTIR. After the previous experiments using resin, glass and intra-lipid we noted that without a Refractive Index matched phantom we could not verify the RDM completely. In order to do this a phantom was constructed using a clear layer of resin to represent the void region instead of the glass. The construction and geometry of this phantom is illustrated in figure 7.34. We note that on this occasion the setup of sources and detectors was in a multiple ring arrangement for the purposes of using the data in another experiment for a reconstruction (see [42]). The set up is a central ring of 8 combined source detector pairs and two off rings of 4 source detector pairs, this gives us the a slightly lesser sampling than in the previous experiments on void validation.

In figure 7.35 we show the Mean-Time of flight from a source to the detectors and the standard deviation thereof in the half ring. It illustrates this in comparison to the computed values for the RDM using a 2mm Gaussian source to match the MONSTIR sources. We see a good match between model and phantom results, there is a

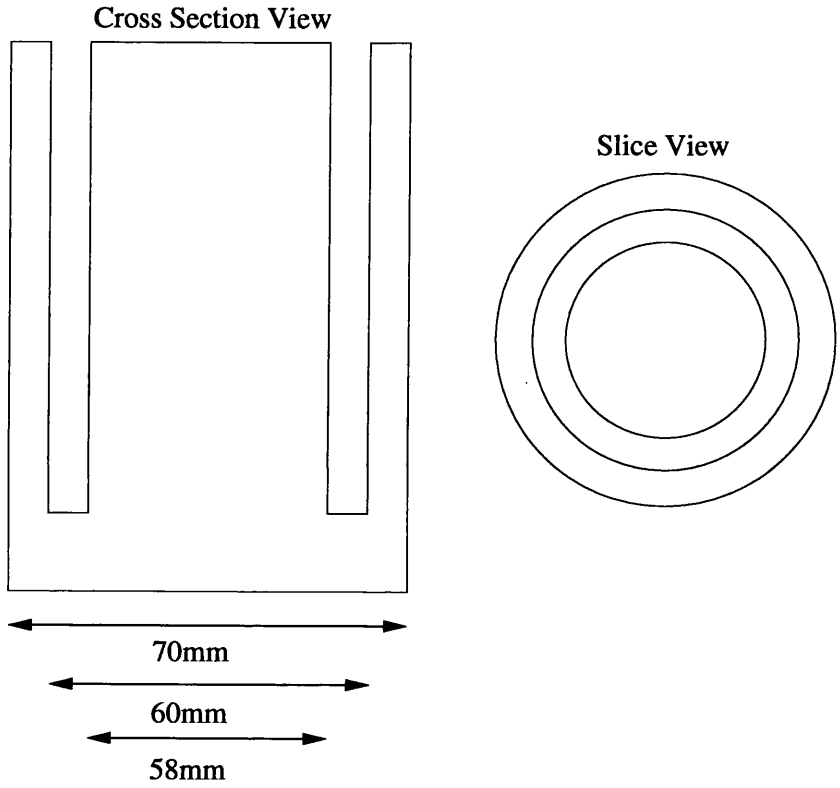


Figure 7.34: A diagram of the Index matched resin phantom showing the location and geometry of the clear layer.

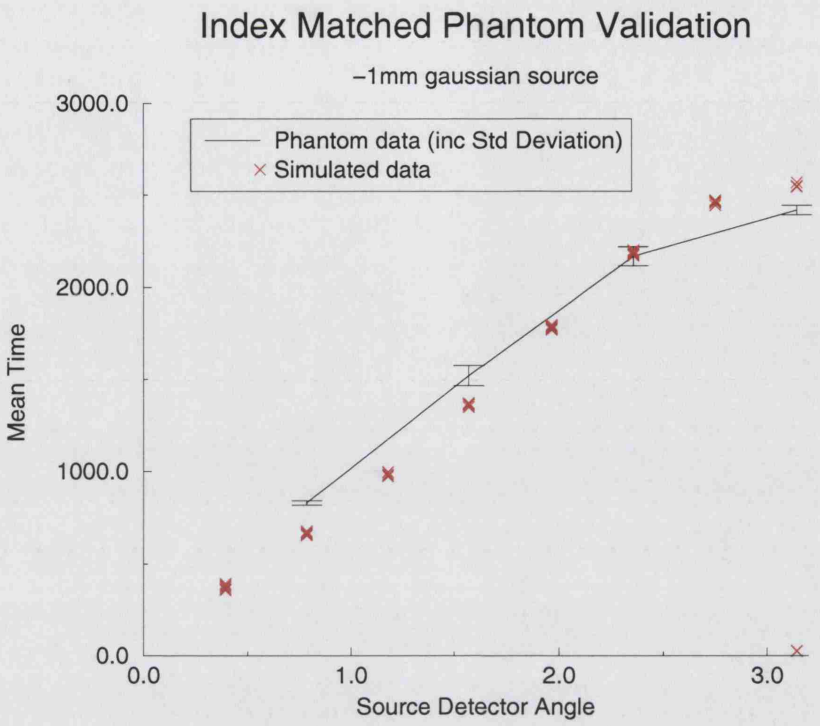


Figure 7.35: A graph showing the mean-time of flight and the standard deviation thereof for the refractive index matched resin void phantom in comparison to the computed RDM values.

slight mismatch which we ascribe to the need to slightly foreshorten the mesh of the numerical model to fit in memory on available machinery.

7.5 Model Comparisons III - RDM vs other models for the domain

As we have already mentioned there have been other suggestions as to how to approach the void problem. Some, such as [46, 73, 74, 90], are based on using a P_1 boundary condition as opposed to a Diffusion based one. These simply suggest using what may be considered a more correct boundary condition, they have associated with them different computational efforts and restrictions. Others suggest using a simpler approximation removing the need for a non-local coupling if the void is a thin layer [13].

We will first compare the diffusive RDM to the P_1 RDM using a BEM against Monte-Carlo Simulation to contrast the two models. Next we will go on to look at the idea of comparing the model RDM against the simplified Modified Diffusion Model

(MDM)[13] and also to compare it with a best fit low-scatter model. The best-fit low scatter is an idea developed after discussion with the one of the authors of the MDM. Here instead of simulating a void with a very low scatter ($\mu_s = 10^{-7}\text{mm}^{-1}$), we use a best fit value. Using this approach we will examine the fields and the boundary data, we go on to compare the shape of the boundary curve in light of results presented in [4] and also to look at the changes in the field produced by an absorbing anomaly.

7.5.1 Diffusion RDM vs P_1 RDM

In this section we compare results between the FEM based Diffusion RDM and a BEM based P_1 RDM. The P_1 model was developed in [73, 75] and implemented in 2D and then extended to a 3D implementation in [90]. Here we show results arrived at in collaboration with Jan Sikora and previously reported in [90]. We illustrate the comparison between the two Radiosity based void models and Monte-Carlo simulations.

We start with an initial comparison in the steady-state case, we show the comparison between the FEM-Diffusion-RDM, the BEM-P_1-RDM and a Monte-Carlo simulation. In each of the simulations we show here, we are using a $25 - 20 - 17$mm concentric sphere void model with $\mu_a = 0.01\text{mm}^{-1}$ and $\mu_s = 1.0\text{mm}^{-1}$ in the diffusing layers.

Figure 7.36 illustrates the match between the three models, it clearly illustrates that whilst the P_1 boundary condition is in fact more correct we get a better fit from the Diffusion based model. The next step is to compare the frequency domain case, here again we use the spherical model as before, but compare the amplitude and phase at 100MHz. We note here that the current level of meshing in the FEM is suitable for a steady-state model, but is numerically unstable near the source in the frequency domain case. However we still see similar results near the tail as for the steady state case for the amplitude (figure 7.37) as we would expect.

More interesting is that neither model matches as well in the phase case (figure 7.38). However we note a better fit for the Diffusion based model near the tail than the P_1 based model.

Clearly these results indicate that whilst the P_1 approach initially seems more correct, the Diffusion RDM is a better solution. There is no direct explanation for this apparent discrepancy, but it is similar to the point collocation method appearing to have

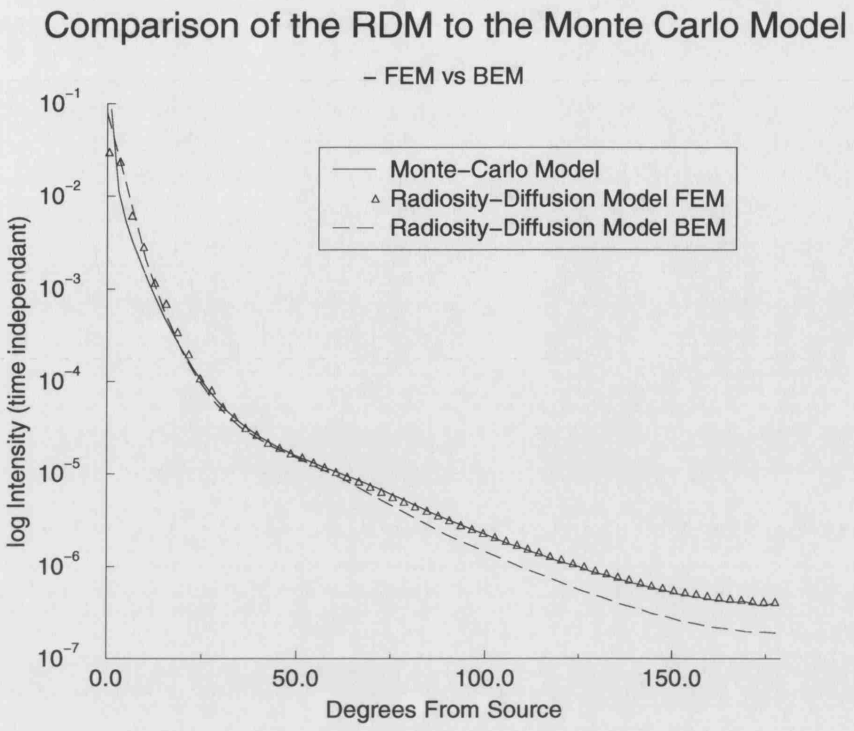


Figure 7.36: A graph illustrating the comparison of the FEM and BEM Radiosity Diffusion Codes to the Monte-Carlo simulation in the steady state - Intensity Data.

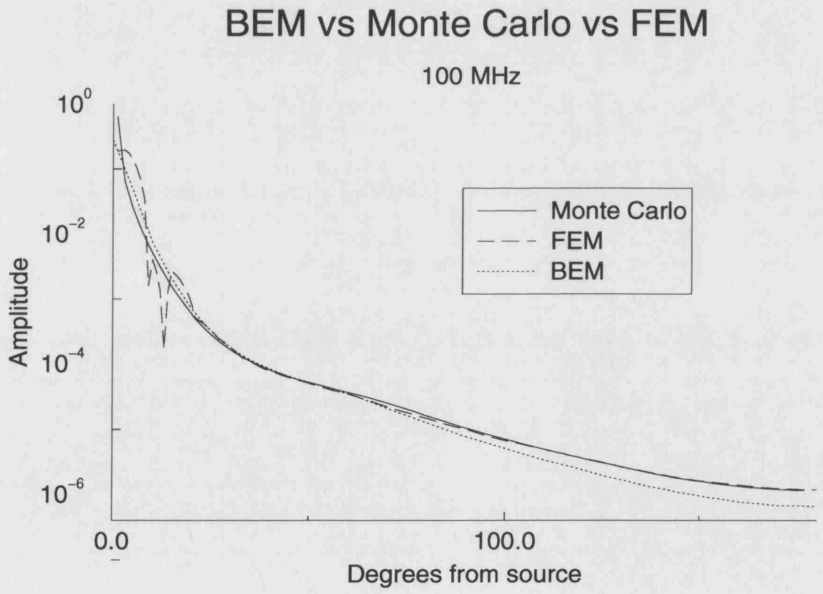


Figure 7.37: A graph illustrating the comparison of the FEM and BEM Radiosity Diffusion Codes to the Monte-Carlo simulation in the frequency case - 100MHz Amplitude Data.

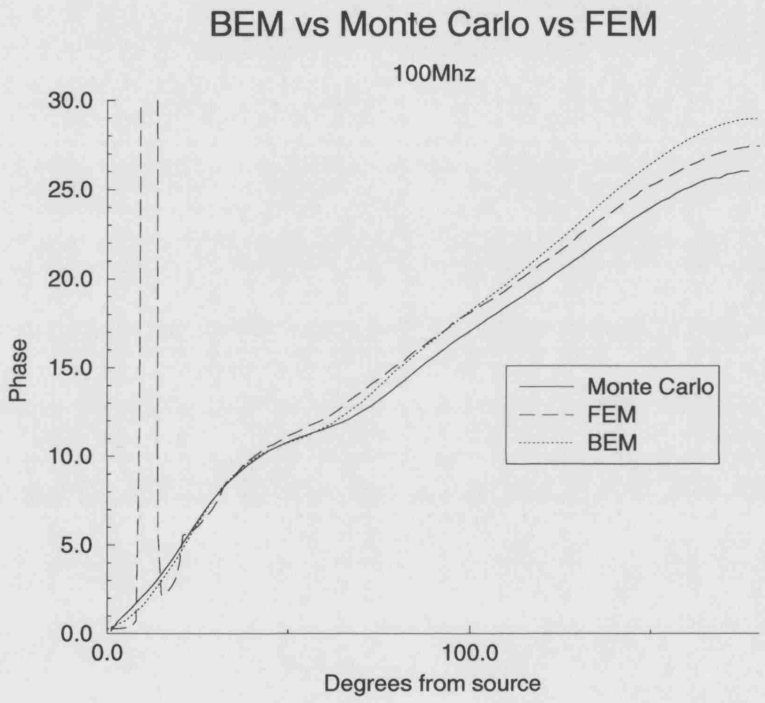


Figure 7.38: A graph illustrating the comparison of the FEM and BEM Radiosity Diffusion Codes to the Monte-Carlo simulation in the frequency case - 100MHz Phase Data.

a better fit to the Form Factor model than the Galerkin, which appears counter intuitive. We suggest this discrepancy is most likely caused in some component of the underlying assumptions being transgressed in the P_1 approach, or that the Diffusion RDM is better suited to our numerical models, as was the case with the Galerkin Form Factor being a better fit in the overall model.

7.5.2 RDM vs MDM vs Low-Scatter

Having established that the RDM is a valid working model we now raise the question of whether it is the most efficient model for approaching diffusing domains containing non-scattering regions. To test this we will compare it to two other models which show promise of being more efficient. The first is the Modified Diffusion Model presented in [13] and outlined in section 5.5.2, the second is an approach, discussed with the author of this work, of using a best-fit low scatter diffusion approximation. We will look at three aspects of the models here, first we will make a visual field comparison to ensure they provide the reasonable internal fields vital to optical imaging, next we will test to see if they are quantitatively accurate, and finally as we know these are local models, we will examine the inflection of the boundary curve caused by the clear layer, which we know caused problems for other models such as the Discrete Ordinates Method.

Visual Field Comparison

The first comparison we undertake is to look at the internal fields and see how they compare to each other. Obviously if we are developing radically different internal field structure our models are inadequate to the problem. In this test case we use the standard $25 - 20 - 17$mm sphere case, with $\mu_a = 0.01\text{mm}^{-1}$, $\mu_s = 1.0\text{mm}^{-1}$ and a refractive index of 1.4. We aim to show that unlike a true low-scatter diffusion these models can at least approximate the field generated by a more correct model such as the RDM, the next section will examine if these fields are quantitatively correct. In the case of the best fit low-scatter DA we are using a selected scattering value of $\mu_s = 0.1\text{mm}^{-1}$.

In figure 7.39 we show the fields for each of the numerical models. We clearly see that whilst the simple Diffusion Approximation and the Diffusion Approximation with a realistic scatter value cannot match the RDM the other two approaches, the Modified Diffusion Model and the best fit low-scatter Diffusion Approximation, do appear to match very well to the RDM. This implies that at least qualitatively we will be able to

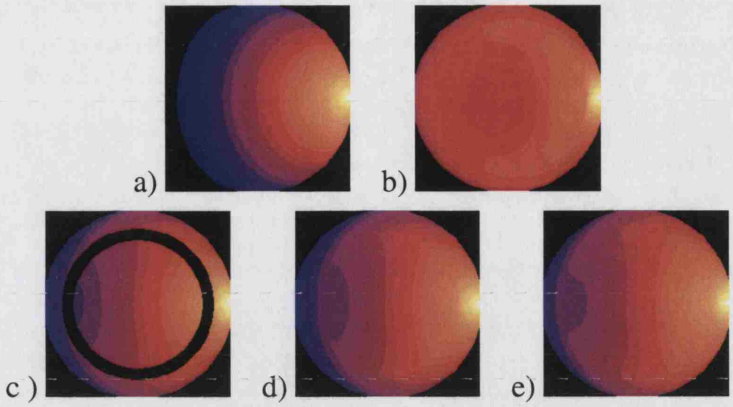


Figure 7.39: Cross-sections in the xy-plane of a $25-20-17$mm concentric spherical void model showing fields for a) a purely diffusive model(ignoring the void), b) a diffusive model using a realistic close to zero scatter ($\mu_s = 10^{-7}\text{mm}^{-1}$), c) the Radiosity Diffusion Model, d) the Modified Diffusion Model and e) a best fit low-scatter diffusive model ($\mu_s = 0.1\text{mm}^{-1}$).

use these simpler models for handling clear layer problems. In the next section we will go on to show a numerical comparison for this and other cases for these models.

Numerical Boundary Comparison

As we have seen above, there is potential for vastly less costly models to simulate our problem to some level of realism. If we are to use these models in a quantitative sense, we must have some numerical comparison to show we really can match the models. To do this we take three concentric sphere models. We note that results given in [13] show only the data far from the source here we will be showing the complete boundary field data.

In figure 7.40 we illustrate the boundary data in the case of a $25-20-17$mm concentric sphere void model, with standard parameters. We note that by setting the DA to have a scattering of 0.1mm^{-1} in the void region we can reasonably match the boundary data. In figures 7.41 and 7.42 we show similar results for a $25-20-18$mm and $25-20-19$mm spherical void model respectively. On examining figures 7.40-7.42 we see that as the void narrows the approximation provided by the Modified Diffusion model improves, as expected.

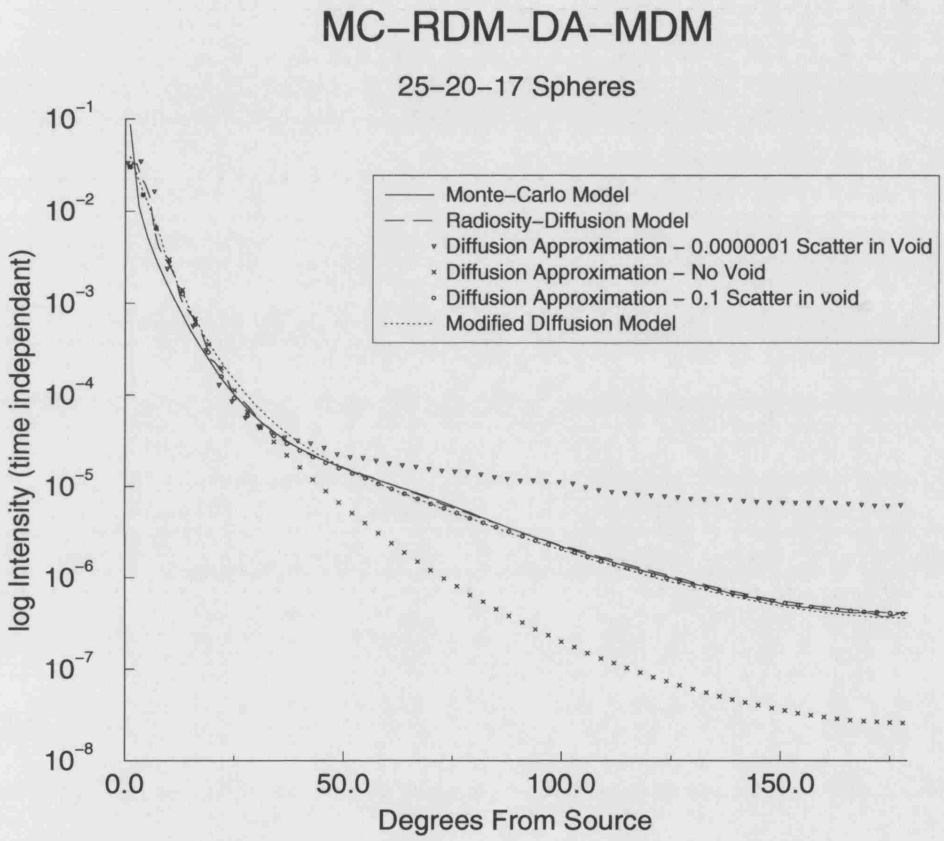


Figure 7.40: A graph comparing the boundary data of a Monte-Carlo simulation with the RDM, the MDM and a best fit low-scatter DA for a $25 - 20 - 17$mm concentric spherical void model.

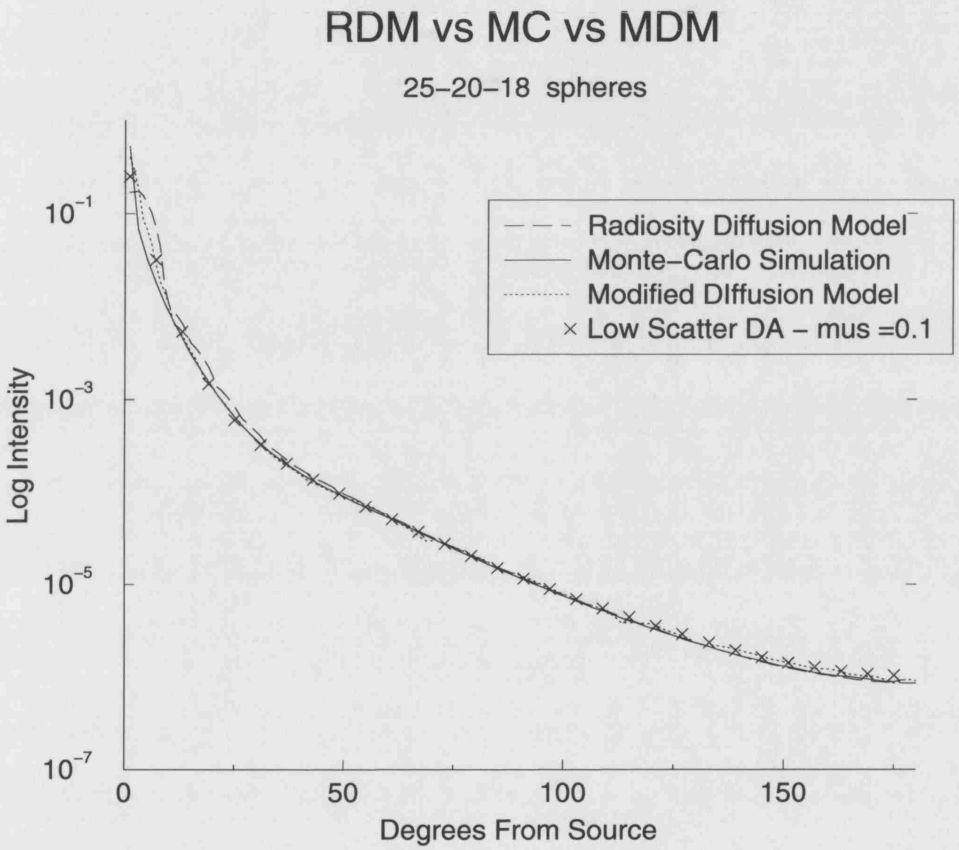


Figure 7.41: A graph comparing the boundary data of a Monte-Carlo simulation with the RDM, the MDM and a best fit low-scatter DA for a $25 - 20 - 18$mm concentric spherical void model.

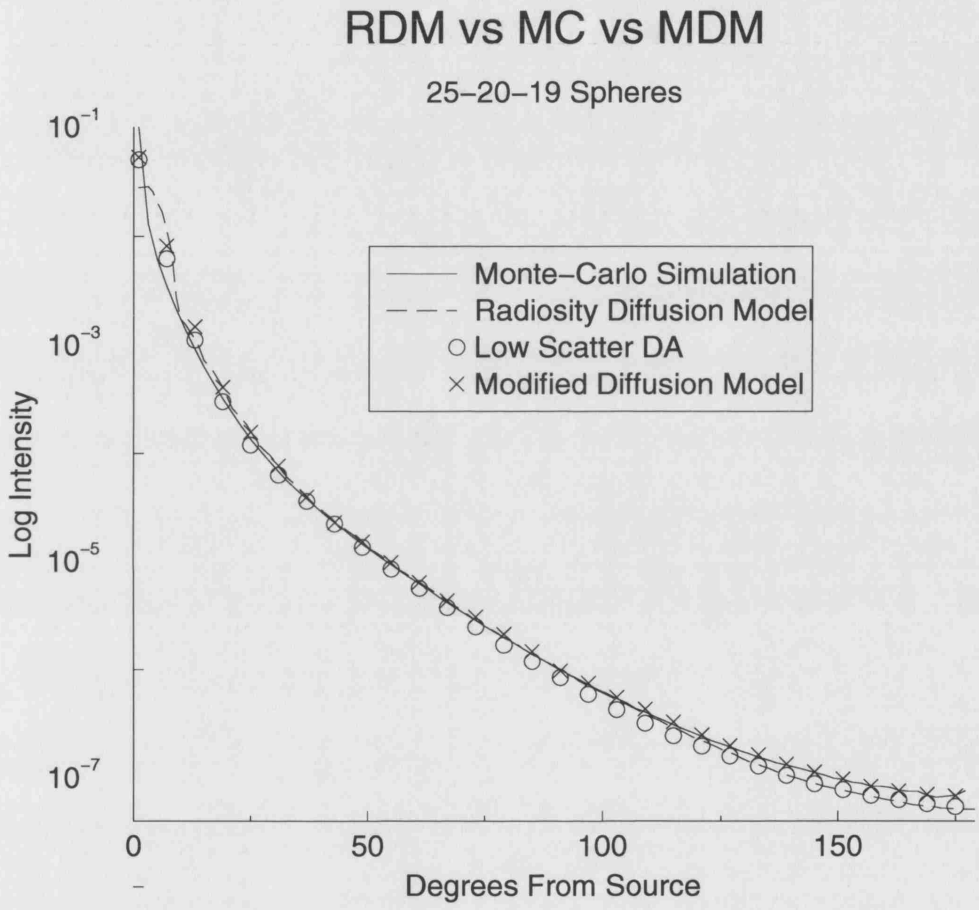


Figure 7.42: A graph comparing the boundary data of a Monte-Carlo simulation with the RDM, the MDM and a best fit low-scatter DA for a $25 - 20 - 19$mm concentric spherical void model.

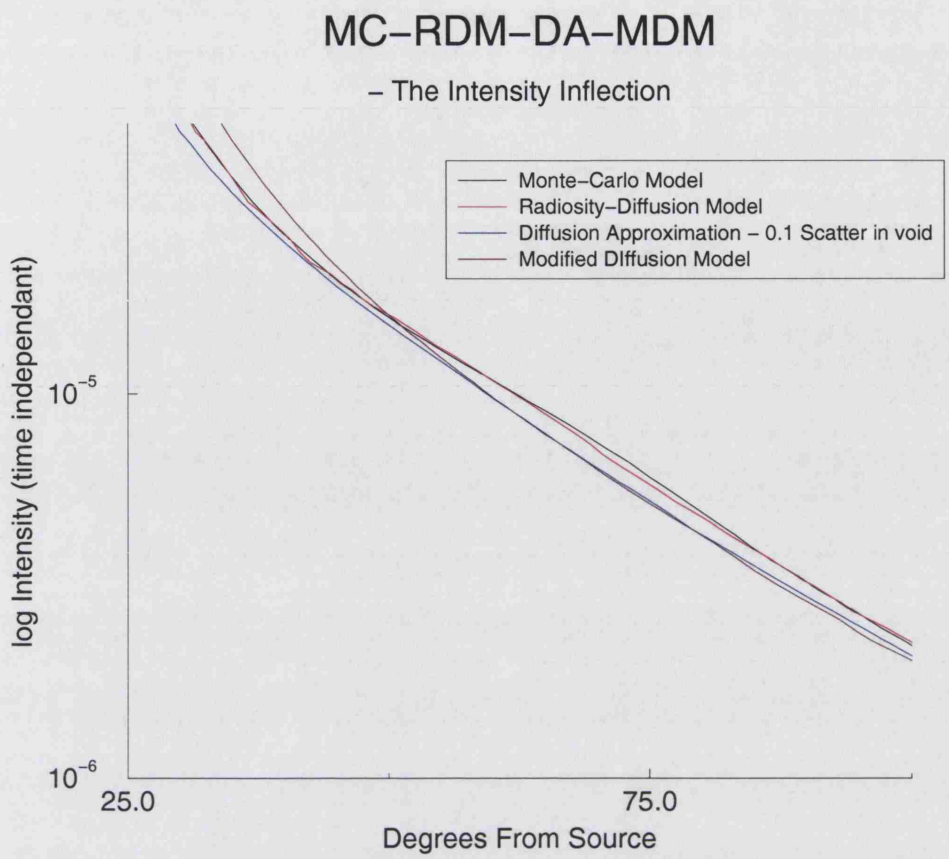


Figure 7.43: A close-up of the graph in figure 7.40 illustrating the inflection present in a correct transport model but absent in the approximations.

The visibility horizon and the Intensity Inflection

In this section we illustrate more clearly the lack of inflection in both the MDM and the low-scatter models. Figure 7.43 illustrates this by showing a close-up of the previous graph (figure 7.40). We note the significance of this in light of results in [4] when considering the discrete ordinates model, which whilst it models the inflection does not do so as accurately as the RDM. As this inflection occurs angularly at the location of the visibility horizon this may prove a significant factor in efforts to reconstruct the void, as from the location at the visibility event, this information has some clear structural significance.

7.6 Application Study - Optical Tomography

In this section we will outline the various effects we wish to investigate in terms of the application of the RDM to Optical Tomography. In [71] we clearly showed that the

3D RDM can be used to reconstruct objects beneath a clear layer, these results will be recapped in section 7.6.1 along with more detailed analysis of the effects of changing the width of the clear layer on the data (initially shown in [72]). In section 7.6.2 we will continue to illustrate the effects of introducing ventricle like void into the model on the data.

We will continue in section 7.6.3 we will discuss the effects of geometric noise on the data. These results initially stemmed from testing the differing visibility models on the data. It transpires that even this minor change from the true inverse crime situation may prevent absolute reconstructions. In this section we will consider the change of the visibility model as a minor variation of geometry and as such illustrate the level of geometric dependence non-scattering inclusions impose for Optical Tomography.

Finally in section 7.6.4 we will move on to examine the effects of using simplified models for the clear layer on the data. We will demonstrate that whilst simpler models can be used to match the boundary data they may provide some difficulties with respect to the imaging of inclusions below the clear layer.

7.6.1 The Clear Layer or Sub-Arachnoid Space

In this section we will deal with the effects of introducing a clear, non-scattering, sub-arachnoid space (SAS) like layer to a diffusive domain. We have already seen that the boundary data is radically different in the presence of a clear layer, therefore we know that we must model this if we are to reconstruct an object behind it in an absolute sense, i.e. from only one data set. What we aim to show in this section is what happens if we were to use the idea of difference data, i.e. a situation where we know a background data set and an anomaly data set. The easiest way to imagine this is the sense of a blob in - blob out experiment. At the end of this section we will illustrate that with a known void we can reconstruct an image from absolute data, how precise our knowledge must be to do this is left to section 7.6.3 for further discussion.

In order to illustrate our problem we first outline our model. In this and following sections we will use a concentric sphere model of radius 25mm with a clear layer beginning at 20mm and of variable width. We will use our standard parameters - $\mu_a = 0.01\text{mm}^{-1}$, $\mu_s = 1.0\text{mm}^{-1}$ and a refractive index of 1.4. Our blob will be a 3mm radius sphere inserted at a position $(12\text{mm}, 0, 0)$ where the sphere is origin cen-

Figure 7.44: An image of the blob used to illustrate the void effect, showing a three slice image through the 3D model with the three slice passing through the blob.

tred, it will have background scattering value and twice background absorption, i.e. $\mu_a = 0.02\text{mm}^{-1}$. We illustrate this geometry in figure 7.44.

In order to get a measure of our data we will use a 32 source and 32 detector arrangement, they will be set up as illustrated in figure 7.45 in three rings around the sphere. We note at this point that this arrangement was chosen as it was similar to the standard experimental set-up of MONSTIR when this work began. Due to modifications in the system co-located sources and detectors are now used. Numerically the principle is the same, but for continuity of our data we have kept the old formation.

There are two ways we can examine the data produced in such a set up, we have 1024 measurements that can be examined qualitatively or quantitatively. The protocol here is that each source is illuminated independently and then each detector has a measure taken for each source - giving us our $32^2 = 1024$ possible measurements. The qualitative analysis is usually performed by looking at the sinogram of the data, a 2D plot of the information. A sinogram is a graphical plot showing a matrix of data values, these are plotted as a chart of sources against detectors. The second option is to examine data in a quantitative sense by plotting it on a graph, looking at 1024 measurements is a lot of data and not very clear, so we tend to plot the measurements for only the first source in our experimental set-up (this is chosen as the source nearest the object and therefore should have the largest data effect). The only other point to note about our measured data is the form the difference data takes. If we look at absolute changes ($Data_{Blob} - Data_{Homog}$) the data is biased by the large magnitude change over the perimeter, so we will use locally normalised data ($\frac{Data_{Blob} - Data_{Homog}}{Data_{Homog}}$) as it provides

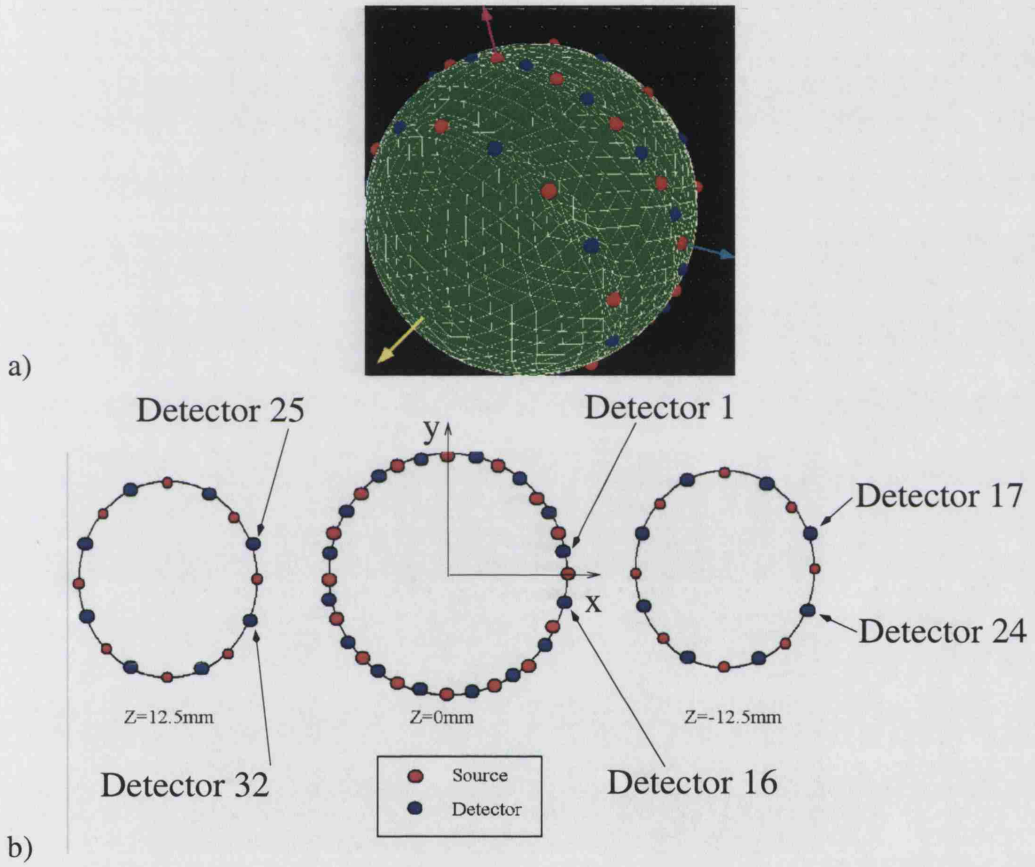


Figure 7.45: A diagram illustrating the source detector layout a) in 3D on the surface (with sources in red and detectors in blue) and b) as a 2D diagram illustrating ordering.

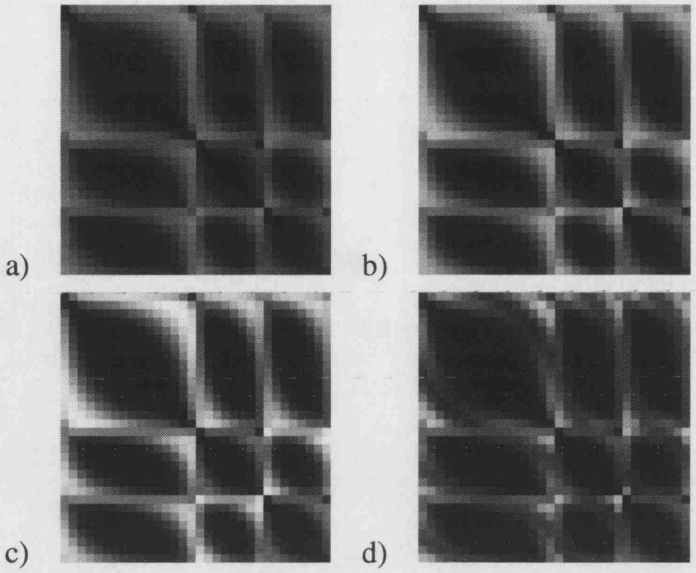


Figure 7.46: Sinograms for the three void layers and the solid with a single blob: a)the 5mm void, b) the 4mm void, c) the 3mm void and d) the solid sphere.

clear content information.

In figure 7.46 we illustrate the qualitative effect of a void on the data. We show the sinograms for four cases, no void, a 3mm void, a 4mm void and a 5mm void. As we would expect the data is blurred as the void thickens, this effect is caused by more light travelling around the void as opposed to penetrating the central region. Also the light which travels around is attenuated less and therefore dominates the data in comparison to that which passes through the central region of interest (in the real case this would be the brain). We also see that the images become darker, i.e. less intense as the void thickens, this tells us we have less information, as we expect for the reasons stated above.

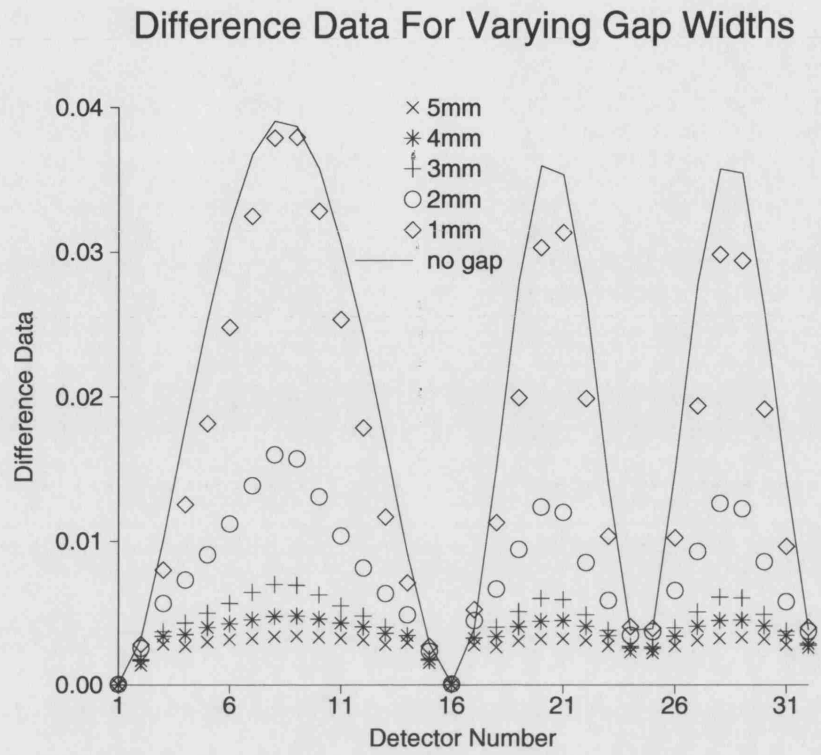
In figure 7.47a) we illustrate a quantitative measure of the effect of a blob on the boundary data for the clear layer case. We show data from a purely diffusive case and for gap widths from 1mm to 5mm, showing the gradual change in the data we would expect. What is important to note is that not only the magnitude of the data is shifting, it is also changing in shape. In figure 7.47b) we illustrate a second effect, that of how the data appears if we use a realistic low-scatter value in a diffusing model of the void. From these images and graphs and graphs it is clear that we might be able to obtain a qualitative reconstruction of an object behind a clear layer with a diffusive model and

difference data, but that such a result would be quantitatively inaccurate. Also as the difference is flattened by the clear layer, we may also expect some problems with the localisation of the blob. We will illustrate these problems at the end of this section with quantitative comparisons of reconstructions of this simple object.

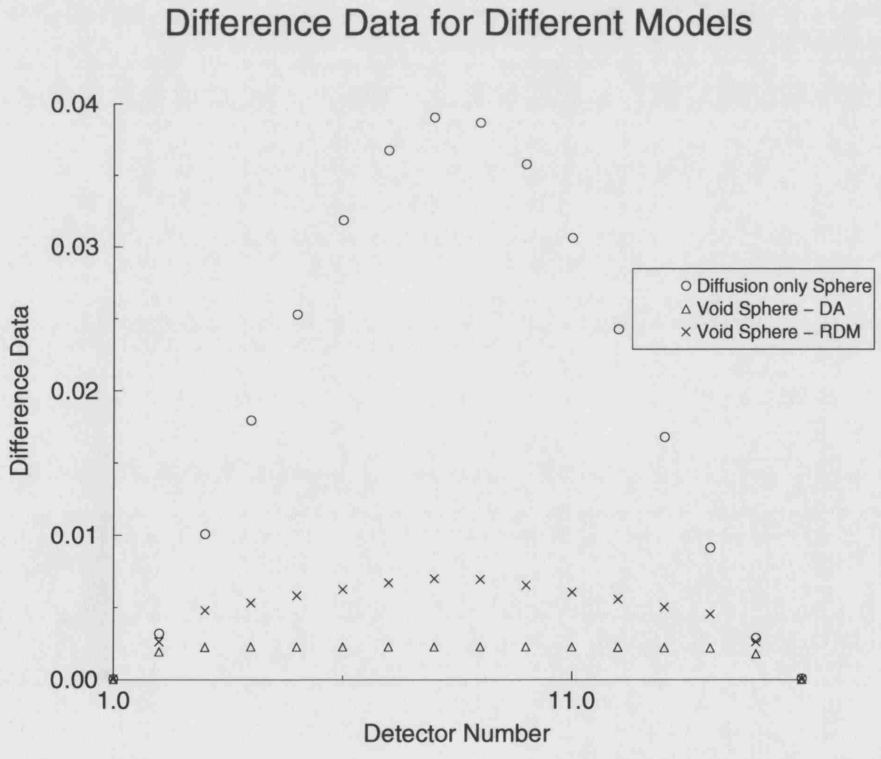
Finally in this section we illustrate capacity to reconstruct objects from data. Here we show the 3mm case first presented in [71]. We show a comparison of target and reconstructed images in three sections taken through the centre of the object. Here we have used an iterative inverse algorithm detailed in previous work from the group [3]. The reconstruction shown uses absolute mean-time data and is an inverse crime case (i.e. we use the same mesh for forward and inverse problems). As we can see we can get a good qualitative result but quantitation is only some 25% of the actual value. Some studies have been done into trying to reconstruct without the void presented in [42]. In these it was shown that in the absence of a void model it is possible to reconstruct a limited amount of information by using difference data, but that as we would expect absolute imaging is not possible without the void being in our forward model.

Having shown the ability to reconstruct this object we move to illustrating the quantitation and localisation problems inherent in the model if we fail to use a void model in the reconstruction. Here we will present three cross-sectional results through the axis of the blob in the above experimental set-up. First we will show the absolute non-linear data acquired in the above experiment, we will then compare this with the difference data result from a model notcontaining a void model (as we cannot use absolute data in this case), to ensure a fairer comparison we will then also represent the same difference data reconstruction on a model containing the void case. In figure 7.49 we clearly see that in the absence of a void model our quantitation is far from providing a reasonable result - in fact it appears flat as it reconstructs the object orders of magnitude smaller than in the void model case.

In figure 7.50 we show a close up of the non-void reconstruction to illustrate the localisation problem. Here we see that as in the void models we have two peaks, however what is interesting is that whilst in the void model image it is clear that the second peak is an artifact of reconstruction and the true objects centre lies close to the true centre of the object, in the non-void case we have two similar peaks appearing just outside the limits of the true object. We note that in the reconstruction images (shown in figure



a)



b)

Figure 7.47: A graph showing the quantitative effect of varying the gap width on the data.

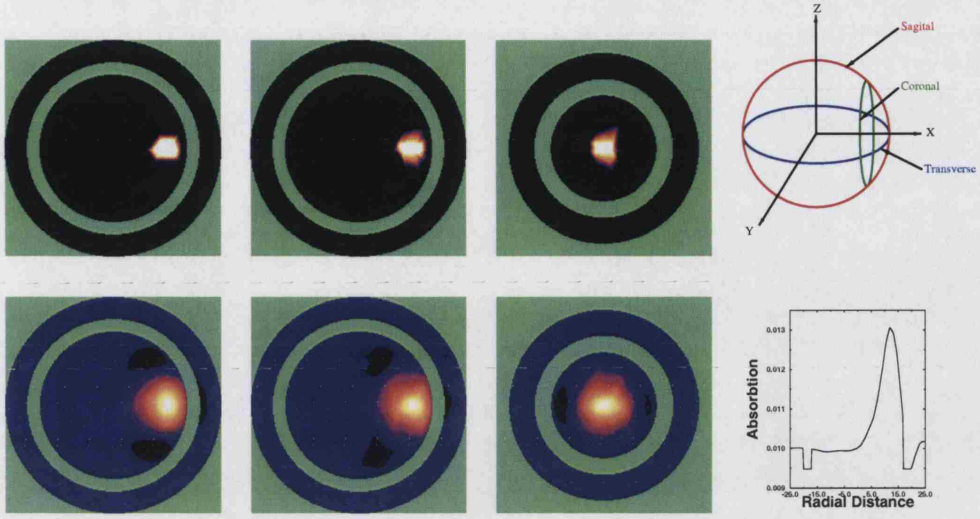


Figure 7.48: Target images (top row) and reconstructions (bottom row) for the $3mm$ gap case. The images are transverse, sagittal and coronal slices through the true centre of the blob, orientated according to the diagram in the top right panel. Bottom right shows a profile along the equatorial diameter through the blob centre.

7.48 for the non-linear and figure 7.51for the linear cases) we can clearly see the localisation failing in the non-void model as we appear to have two objects reconstructed, whereas in the void models we clearly see the second peak is a shadow caused by the presence of the void.

7.6.2 Hollow Regions or Ventricles

In this section we will deal with the effect of clear inclusions equivalent to ventricles on the data. The significance of this is in how it effects our ability to use simplified models. If for instance the ventricles have little or no effect on the data we may be able to use simplified models such as the MDM to solve for the optical parameters in the case of OT on a neo-natal head. If the effect of these ventricles is significant however we will have to use a model such as the RDM unless we can find another simple approximation for such inclusions.

To illustrate the effects we have taken a basic spherical head model, as illustrated in figure 7.52. This is a standard 3mm void concentric shell containing two 3mm radii ventricles located along the y-axis. This allows us to perform two tests on the effect of the ventricles by moving the blob in and out along the x-axis (figure 7.53b) or by

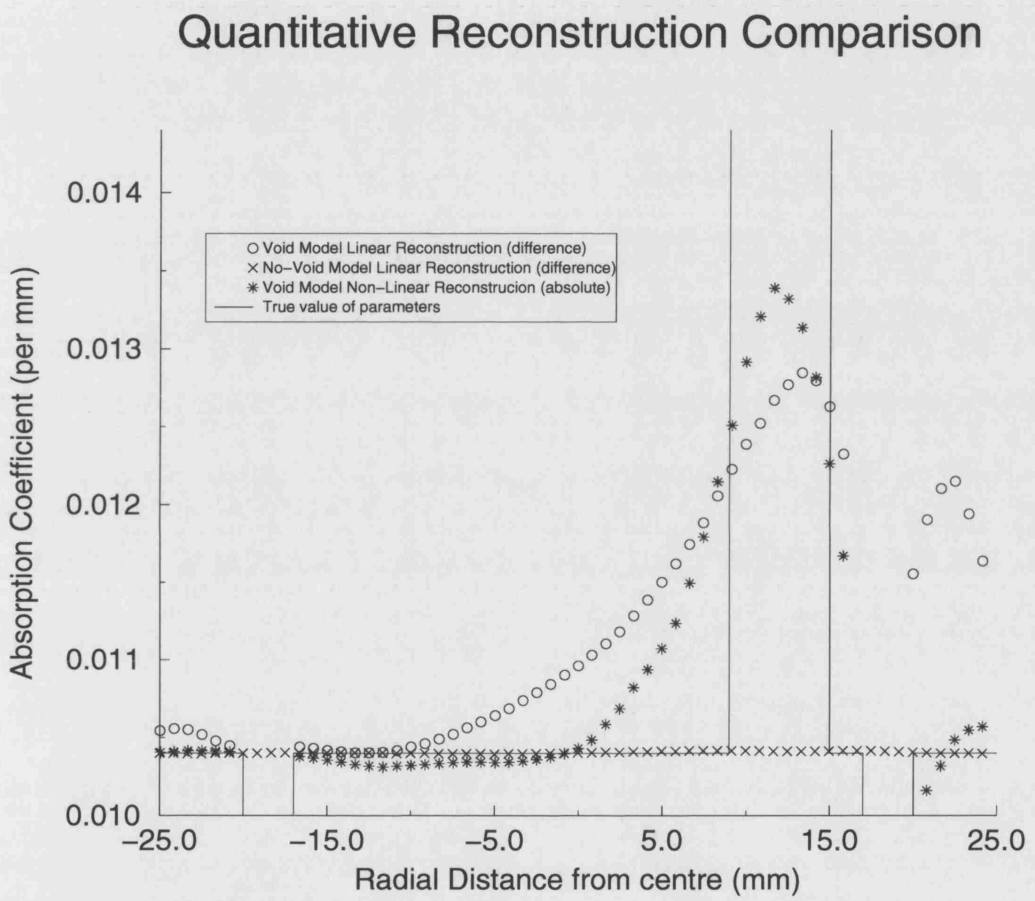


Figure 7.49: A graph showing the quantitative comparison of the void model versus the non-void model reconstructions.

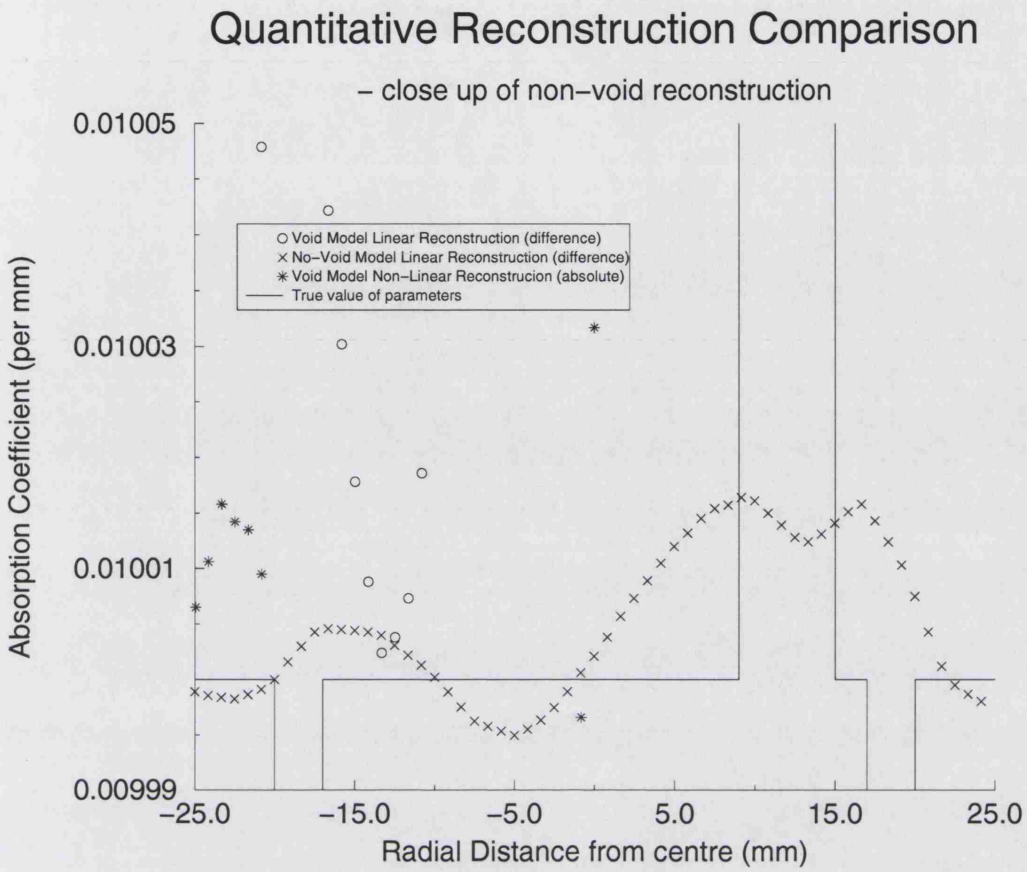


Figure 7.50: A close up of the graph in figure 7.49 showing the poor localisation of the non-void reconstruction.

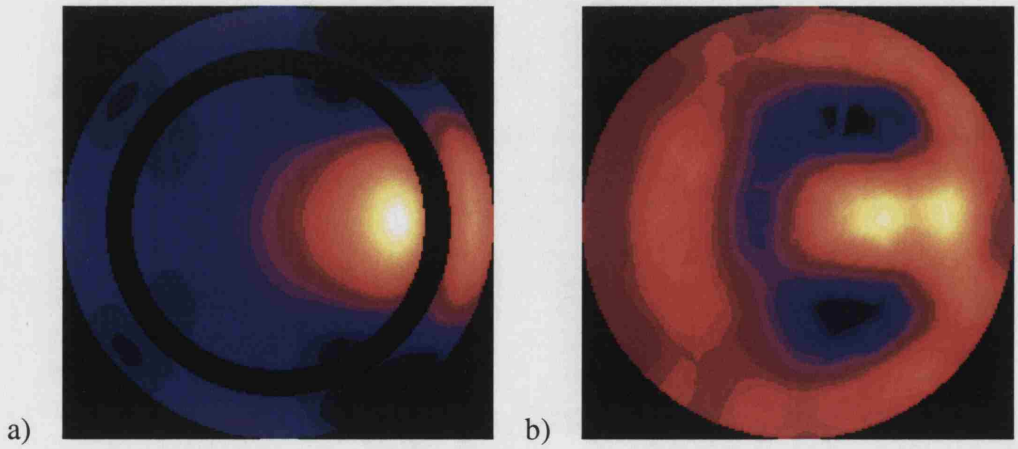


Figure 7.51: Two figures showing the qualitative reconstructions given by difference reconstruction a) with a void model and b) with a non-void model.

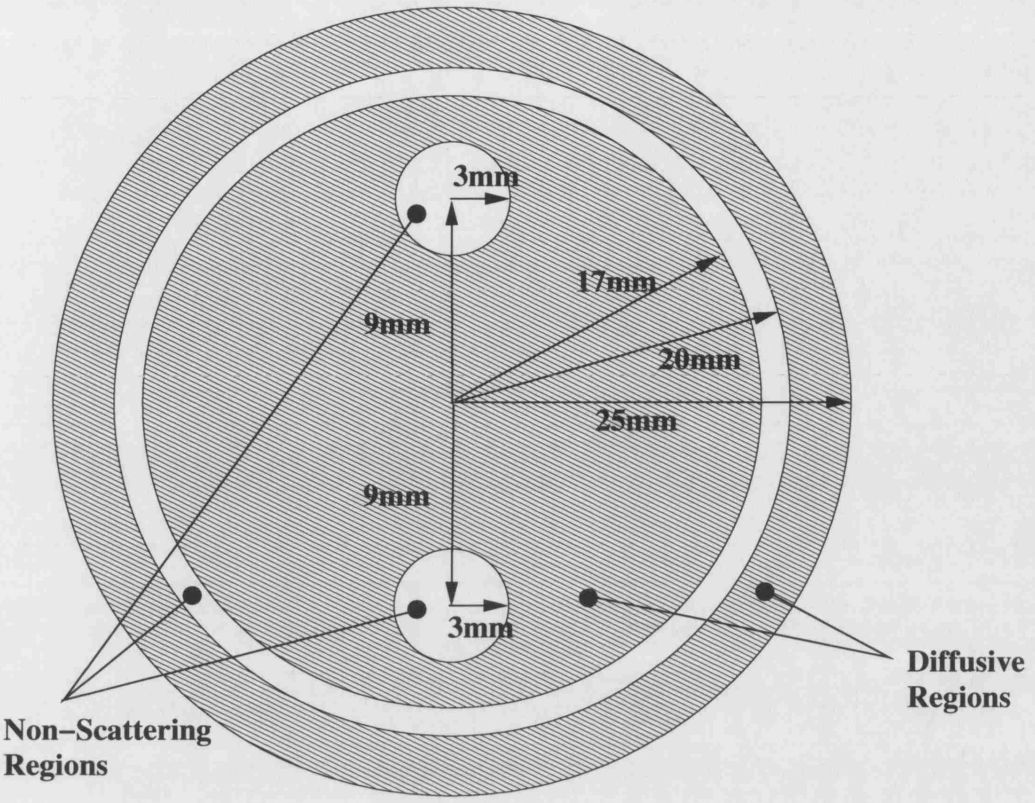


Figure 7.52: A diagram illustrating the geometry of the spherical brain used.

rotating the ventricles with respect to the blob (figure 7.53a).

Illustrating the field differences

Initially we will show results for the field patterns for a ventricle model, both as a qualitative field image and as a quantitative boundary measure. We will clearly see that the ventricles do effect the field significantly. We would expect therefore to see strong results in how they effect the data when we introduce a blob. In figure 7.54 we illustrate the field pattern in terms of the visual field intensity in the medium. It clearly varies from what we get if the ventricles were not present, the question becomes how far from the ventricles do the distortions have effect.

In figure 7.55 we illustrate the data at the boundary. We clearly see a vast difference between the two fields as we would expect. This difference is significantly larger than the effect of a blob in the model, it clearly represents a requirement to model the ventricles in any situation where we hope to provide absolute imaging. In the next section we will discuss the actual effect of data in terms of whether ventricle modelling will be required for difference images.

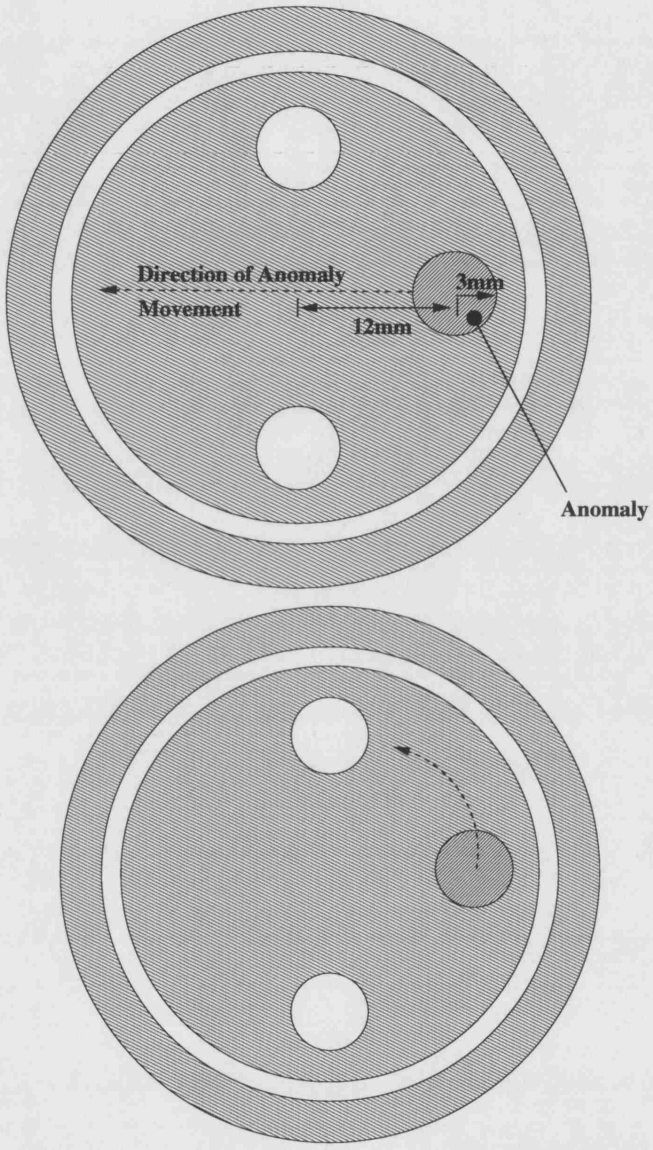


Figure 7.53: A diagram illustrating the two blob trajectories used to test the effect of the ventricles, a) translation between the ventricles and b) rotation with respect to the ventricles.

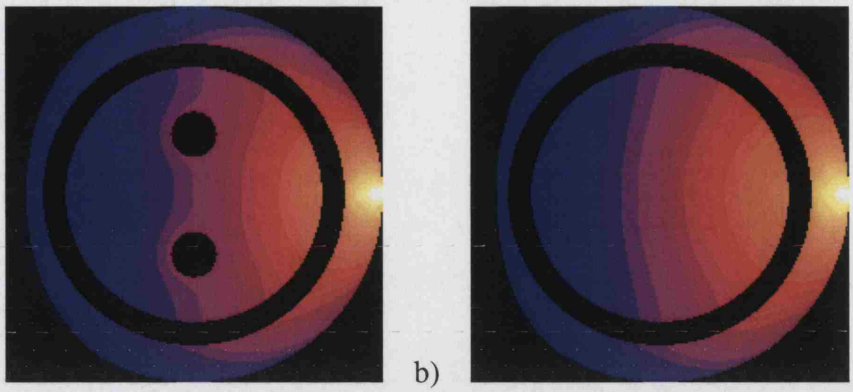


Figure 7.54: A field plot of the logarithmic light intensity inside a simple spherical head model, a) with ventricles and b) without

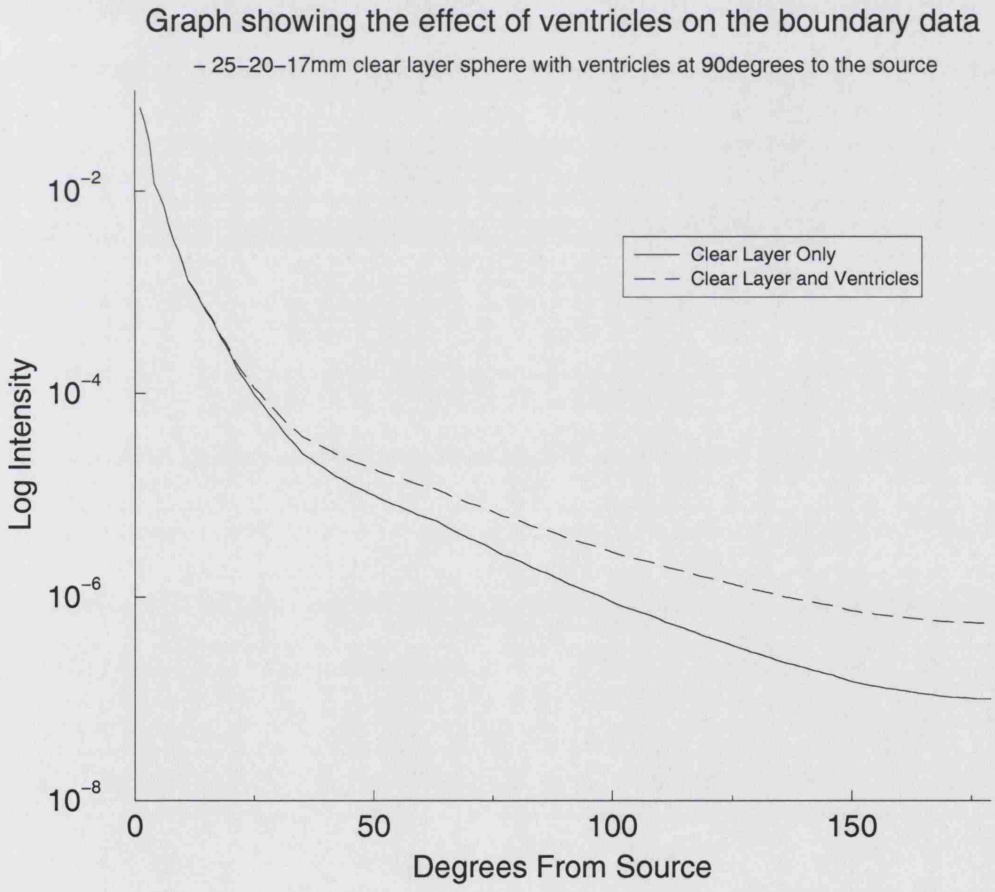


Figure 7.55: A graph showing the effect of the introduction of ventricles on the boundary data of a simple spherical head model.

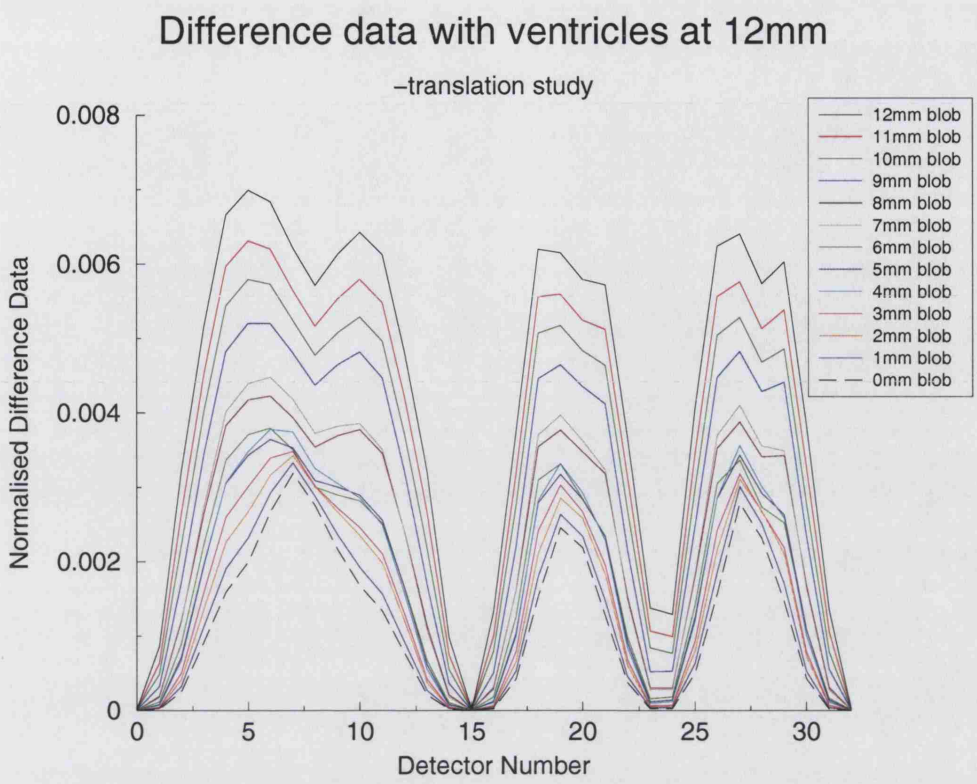


Figure 7.56: Normalised difference data for a blob translating between ventricles situated at 12mm from the origin on the y-axis.

Effect on the measurable data

In this section we will illustrate the effect of a blob moving in translation inside the model as illustrated in figure 7.53a. We do not illustrate the effects of the rotational movement of the ventricles with respect to the blob as the data proved to be unclear due to numerical instability caused by a rapid changing of support of the blob - an effect caused by meshing artifacts. In the translation case there is some effect caused by this, but the general trends in the data are still visible.

In figures 7.56-7.58 we illustrate the effect on the normalised difference data for two translations. In each graph the ventricles are fixed but the blob translates from near the surface of the void (centred at $(12\text{mm}, 0, 0)$) to the centre of the spheres (centred at $(0, 0, 0)$). The three graphs show a three step translation of the ventricles being centred at $(0, \pm 12\text{mm}, 0)$, $(0, \pm 9\text{mm}, 0)$ and $(0, \pm 6\text{mm}, 0)$.

The graphs give us two pieces of information regarding the effect of the ventricles. First we can see in each case two peaks rather than one in the data. This suggests that

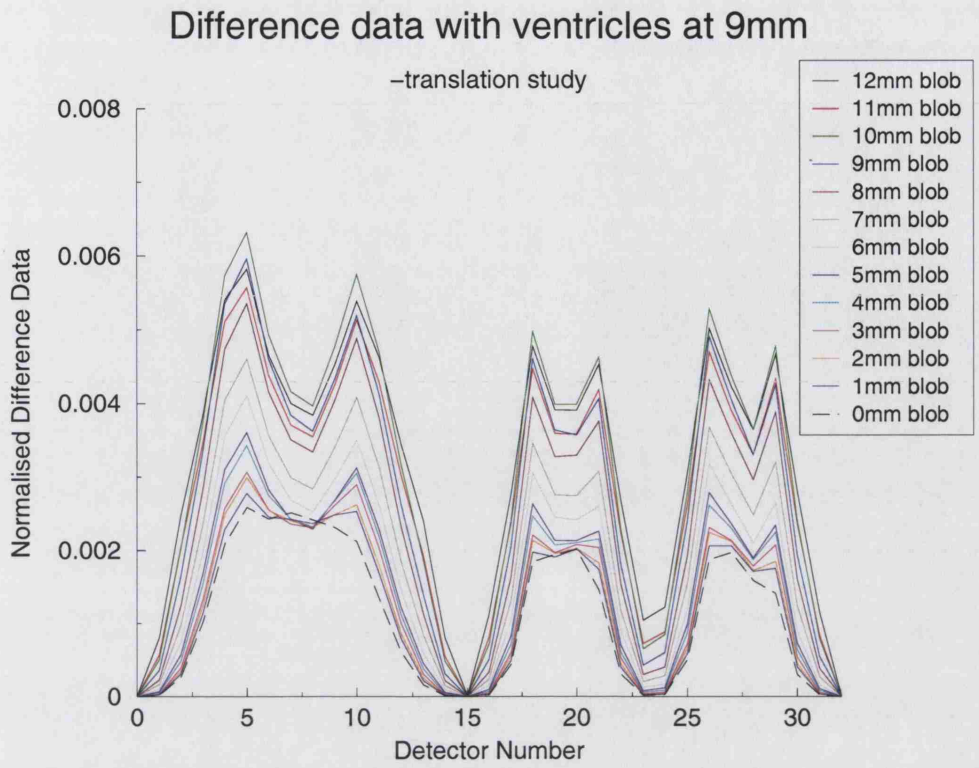


Figure 7.57: Normalised difference data for a blob translating between ventricles situated at 9mm from the origin on the y-axis.

we might expect more than one object to appear in a model where we have no ventricle model. This is significant as it implies that we must model the ventricles in order to obtain even a qualitative image from difference data. This was not necessarily the case with the clear layer and is therefore a significant result when considering optical tomography of the neo-natal head.

The second result is equally striking, it is often commented that as the ventricles are deep feature their effect on the data will be small as we know there is little light penetration at that depth. However we note that whilst the effect on the boundary of deep ventricles is smaller in the absolute sense, the effect appears stronger in the data. That is we have more effect from deeper ventricles on difference data than we do from shallower ventricles.

7.6.3 Geometric Noise

In this section we will consider a significant issue in the Optical Tomography problem. If we do not exactly know our geometry can we find objects beneath the clear layer.

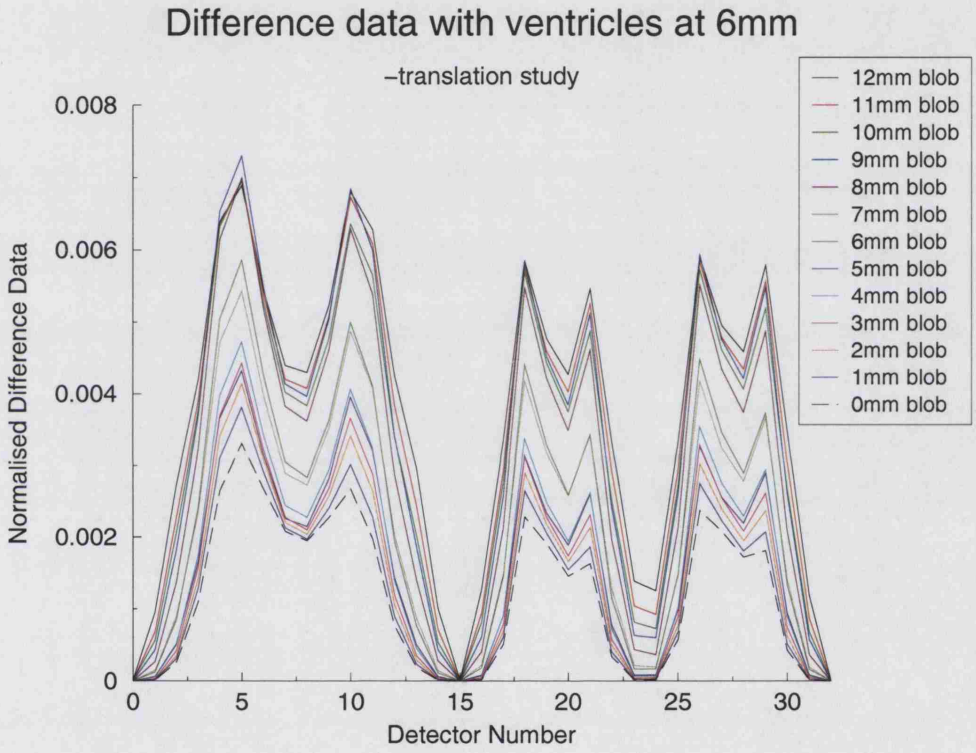


Figure 7.58: Normalised difference data for a blob translating between ventricles situated at 6mm from the origin on the y-axis.

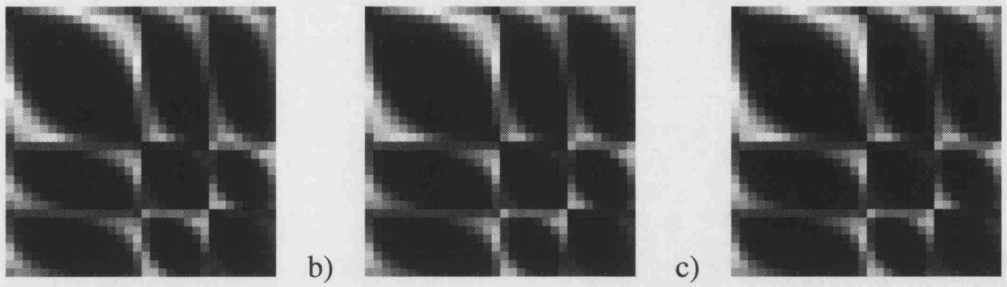


Figure 7.59: Sinograms of difference data for the standard blob in a $25 - 20 - 17$mm shelled sphere using the same visibility model for homogeneous and blob data, a) Analytic Visibility b) OpenGL Visibility c) Voxel Visibility.

This question is important for two reasons, first in that it is likely that we may not know the internal geometry of the head (relating to the location of say the void regions) and may need to reconstruct that as well as the optical parameters. Second, much of the literature regarding the question of need for a void model has focused on the use of difference data, so we would like to know if approximate geometries are sufficient in this case. Here we are asking if using some generalised head model would allow us to handle void data in a difference data reconstruction.

As we have already seen the effect of the visibility model can be considered as a very slight blurring of the geometric information. Particularly if we consider the difference between analytic and voxel visibility which comes down purely to the error produced by discretising our domain. Also if we consider the magnitude of the dimensions of this error it is in the order of $<$ 1mm. This is essentially as good a prior as we may expect to get. So the first step in looking at geometric noise is to compare the effect of the visibility models on the data.

The first test is illustrated in figure 7.59, here we illustrate that the qualitative response of a blob under the three visibility models is the same. That is we have taken normalised difference data using the same visibility model in each case and generated the sinogram of the data for each model. Note: again we are using the 25-20-17mm concentric sphere model with a twice absorbing 3mm radius anomaly inserted at (12mm,0,0). We can clearly see that each model is producing consistent results in a qualitative sense, which implies a good agreement between the models as expected from the stability of the RDM.

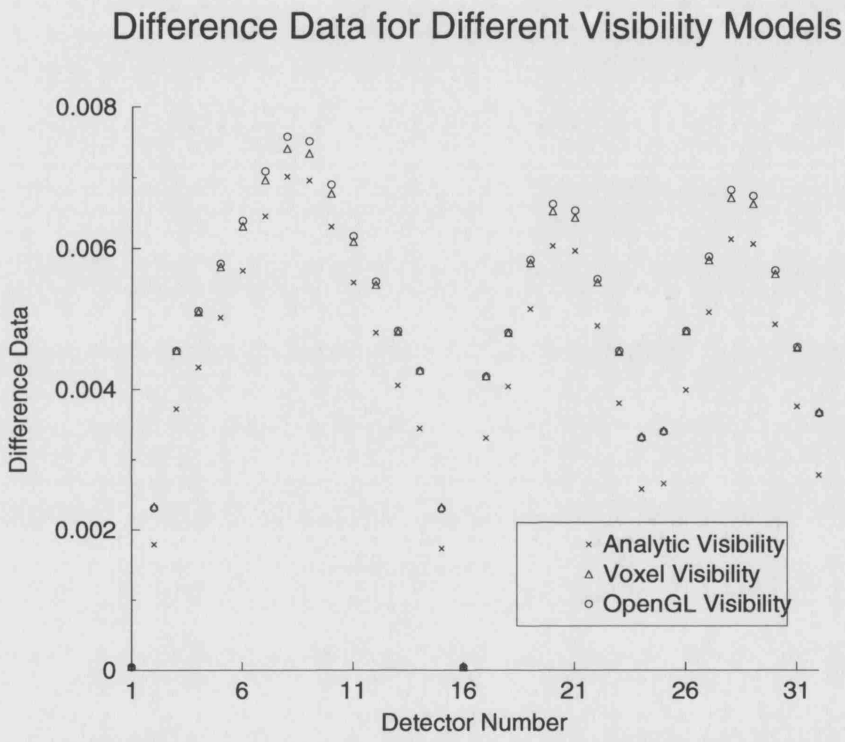


Figure 7.60: A graph illustrating the quantitative comparison of difference data from the three different visibility models, using the same model for blob and homogeneous data.

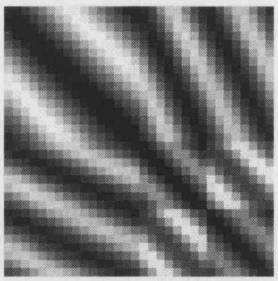
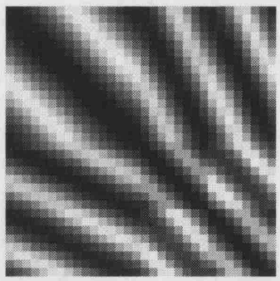
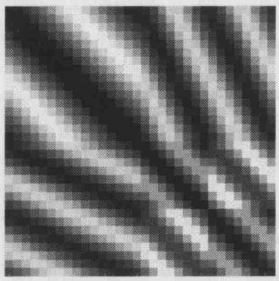
a)  b)  c) 

Figure 7.61: Sinograms of difference data for the standard blob in a $25 - 20 - 17$mm shelled sphere using the same visibility model for homogeneous and blob data, a) Analytic Visibility - OpenGL Visibility b) Analytic Visibility - Voxel Visibility c) OpenGL Visibility - Voxel Visibility.

If we further examine the quantitative results, supplied in figure 7.60, we see that the three models do in fact agree very well numerically as well. This however is a preliminary to examining the geometric noise, we have only established that the three models do in fact behave consistently with respect to changes in the model, we have not as yet introduced the noise element. In figure 7.61 we present the qualitative data for the case where we use a different visibility for the background as for the foreground. This information is essentially what is available to an inverse solver in the absence of an inverse crime or difference data.

We clearly see that the image has become significantly blurred from those seen in figure 7.59. In order to compare this better we examine the quantitative results for a single source as seen in 7.62. This provides a disturbing image as while the shape of the graph looks the same the quantitation is out by an order of magnitude. If we examine the graph in figure 7.63 we can see that if we just compare the data produced by using a different visibility model as our blob we get noise much greater than our data. This is of deep concern, in order to understand this we must examine the boundary data more closely.

As we have already seen the boundary data for the three models does look reasonable. Here we reproduce the three boundary data sets using different visibility models but add in the three sets with the absorbing anomaly present as well, see figure 7.64. What we would hope to see is two clusters of three lines, indicating the effect of the blob is greater than that of the visibility model. However we see a single grouping.

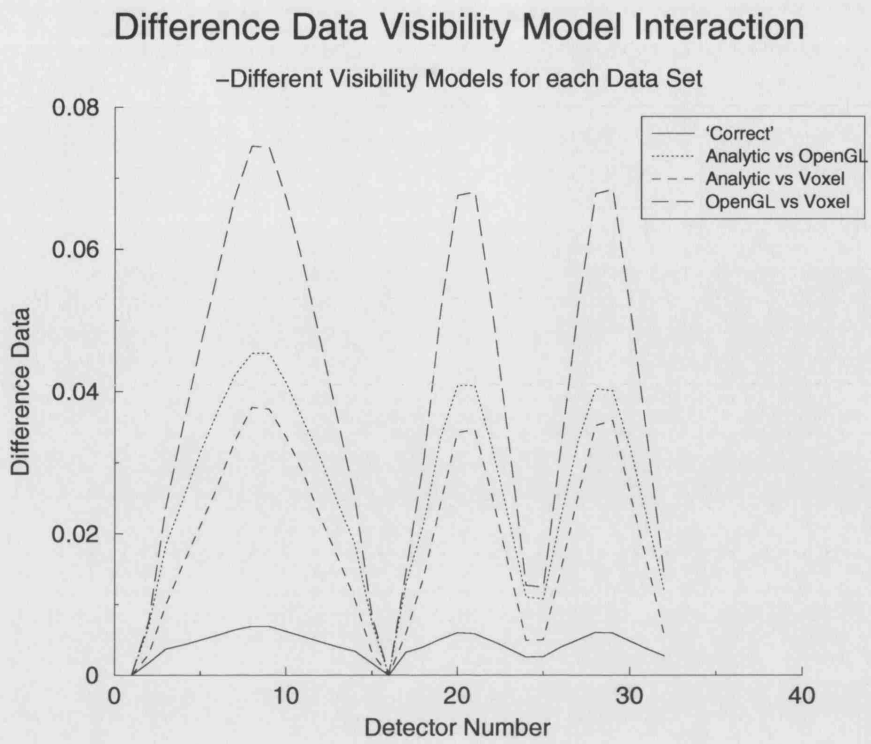


Figure 7.62: A graph illustrating the quantitative comparison of difference data using different visibility models for each data set.

To get a better idea we look more closely at the graph in figure 7.65. Here our worst suspicions are confirmed, we have three clusters of two lines. This means the effect of varying the visibility model, equivalent to a minor geometric uncertainty has out weighed the effect of the blob, or the information in the image.

The implication here is that we cannot hope to reconstruct such an object in an absolute sense in anything other than an inverse crime model, unless we can regularise for the different effect produced by the geometric variation - in essence unless we have difference data. It is important to consider that we have two factors to consider here, first this is a thick gap and second the blob is small. In order to get an idea of what kind of data we can get from different blobs we present data from varying the optical size of the blob (either by changing its absorption property or its physical size) and from reducing the gap width.

In figure 7.66 we present the results of varying the size and absorption contrast of the blob under the 3mm clear layer. We do not vary the positioning of the blob, i.e. its centre at (12mm,0,0), but we vary its radius from 3mm to 5mm and its absorption

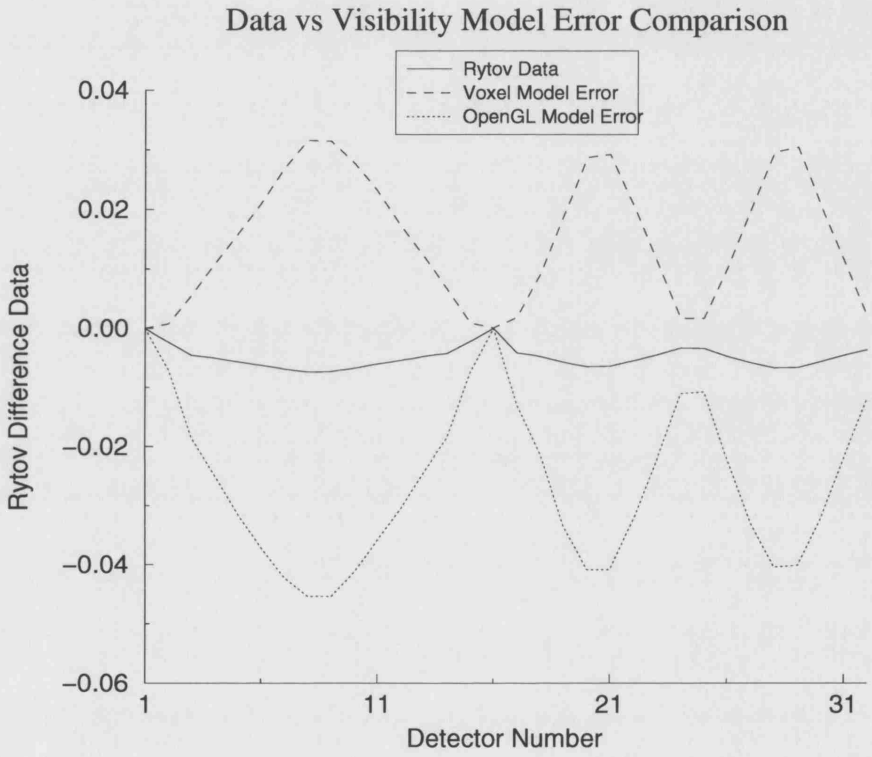


Figure 7.63: A graph illustrating the relationship between signal and noise in the use of different visibility models.

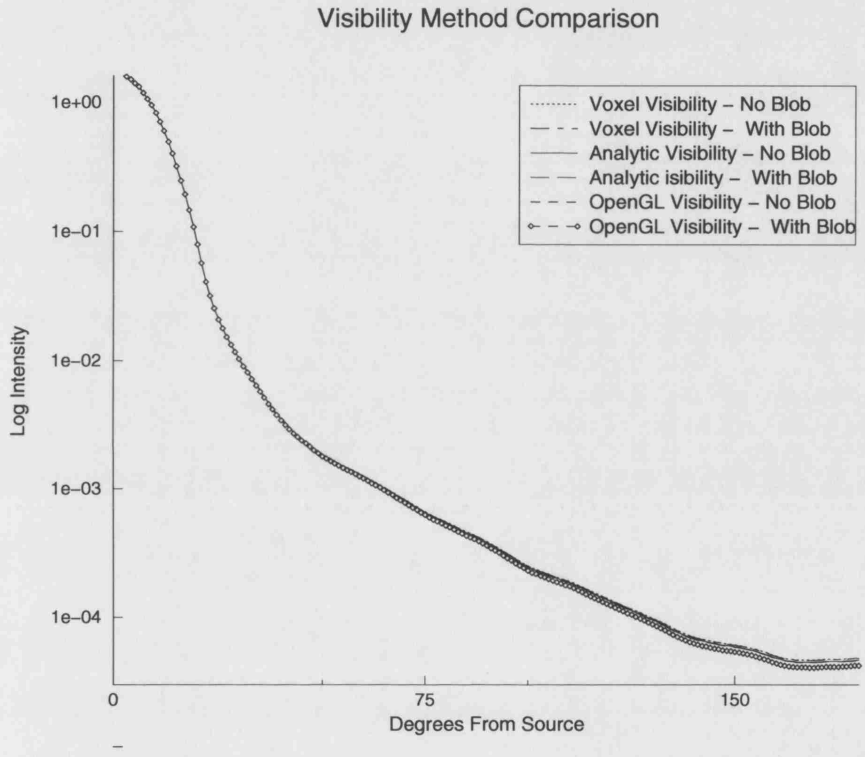


Figure 7.64: A graph showing the peripheral boundary data for the blob and homogeneous models for all three visibility models.

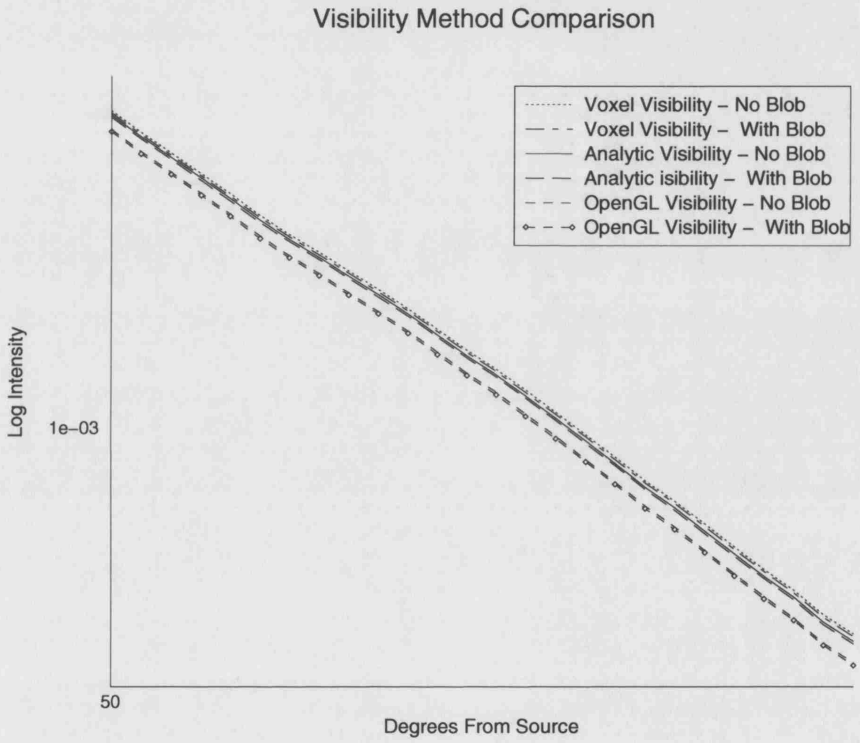


Figure 7.65: A close-up of the graph in figure 7.64 showing why the difference data fails using different visibility models.

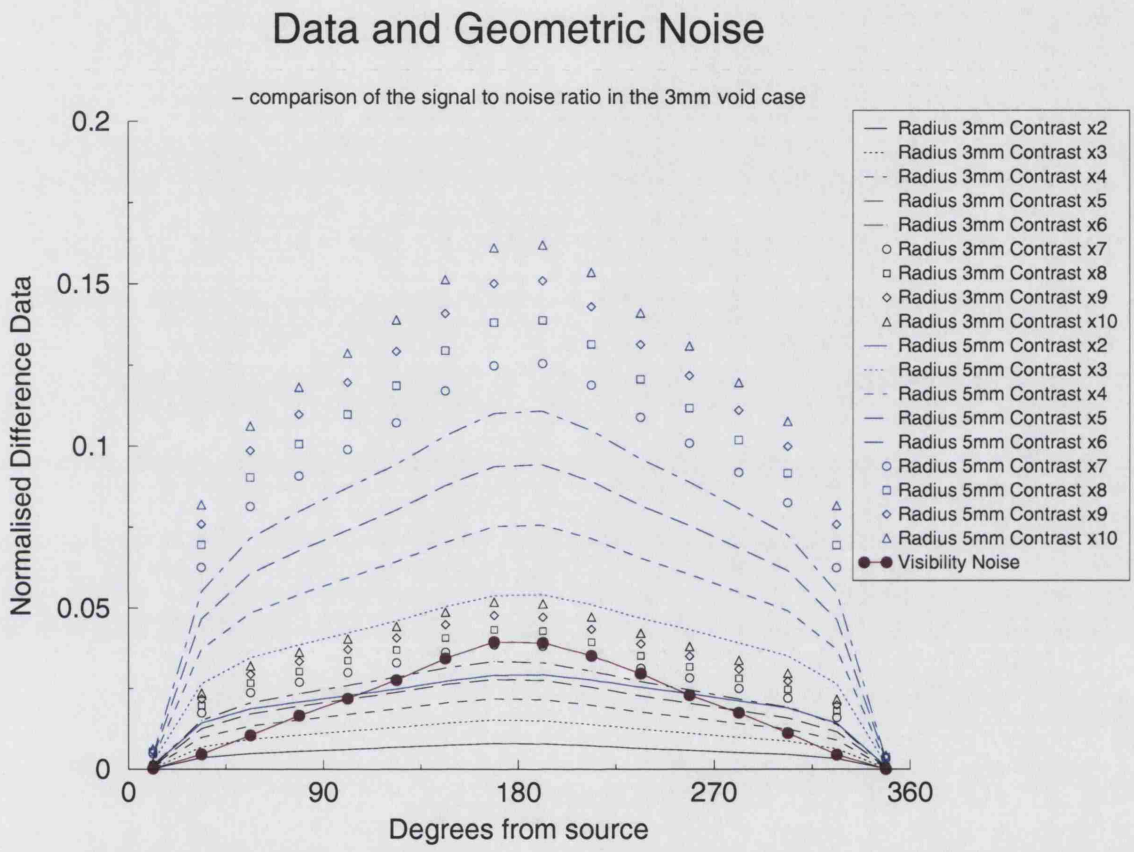


Figure 7.66: A graph showing the signal to noise ratio of different blobs against the noise produced by different visibility models in a 25-20-17mm concentric sphere void layer model. The blobs are given by their radius and their contrast to background absorption, all are centred at (12mm,0,0) and the source shown is at (25mm,0,0).

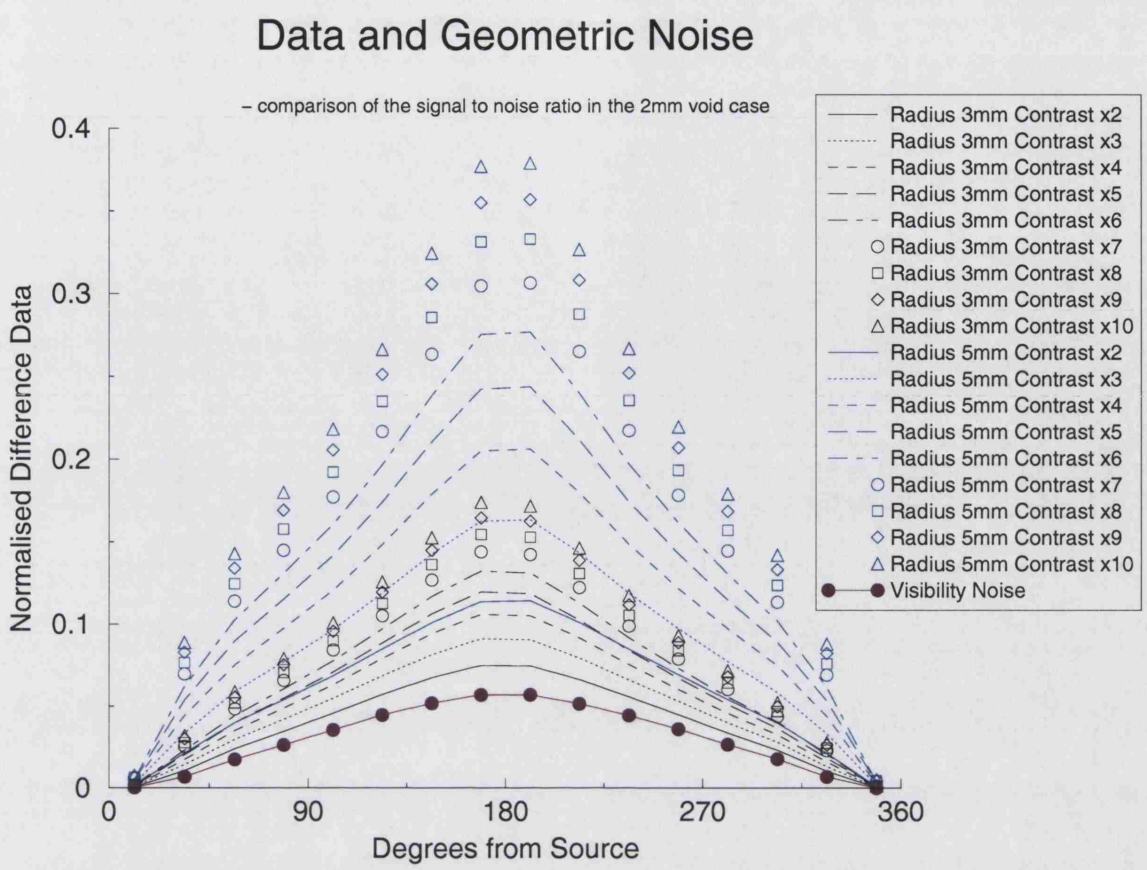


Figure 7.67: A graph showing the signal to noise ratio of different blobs against the noise produced by different visibility models in a 25-20-18mm concentric sphere void layer model. The blobs are given by their radius and their contrast to background absorption, all are centred at (12mm,0,0) and the source shown is at (25mm,0,0).

contrast from $2\times$ to $10\times$ background absorption. In figure 7.67 we show the same information but for a 2mm gap, i.e. a 25-20-18mm concentric clear-layered sphere.

We see here that the effect of geometric uncertainty in the clear layer will be sufficient to suppress small objects from being seen in an absolute sense. It is clear that with only one data set we will not be able to see small objects behind the clear layer, we will also have significant noise generated by the geometric uncertainty itself. We clearly see however that this problem will reduce as the gap grows thinner, although the noise remains at a similar level in the 2mm gap to the 3mm gap, the signal from the objects has become greater due to the reduction of dampening caused by the clear layer. We will not in this thesis predict an actual size of optical object visible beyond

the sub-arachnoid space in a neo-nate as this will wholly depend on the individual case and may further depend on other factors. We have simply illustrated here that as geometric problems in modelling clear layers may interfere with our ability to image small objects absolutely, this may effect our decision as to whether we should use a difference imaging approach.

7.6.4 Comparison of changes in the medium in the RDM, MDM, and the Low-Scatter DA

One thing we will need to consider with such models is if they can sustainable reproduce data for the clear layers. By this we mean that if we vary the parameters of the underlying medium do we get the same response in our data. It is all very well to produce a fit to one medium, but do variations in that medium behave as we would expect. To illustrate this we will investigate the effect on the boundary data of inserting a sphere of double background absorption of radius 3mm with a centre position $(12\text{mm}, 0, 0)$ directly beneath the source. If we then compare normalised difference data on the boundary $\frac{Data_{Blob}-DataHomog}{Data_{Homog}}$ we see the results illustrated in figure 7.68. Here we clearly see that whilst the MDM provides similar data to that we get from the RDM, the low-scatter Diffusion Approximation approach fails to realistically match the data. This indicates that whilst until now we may have been able to consider the low-scatter diffusion approximation a reasonable approach to the problem, not only must we find our best-fit value, but we must also be aware that the response to change is different and would lead to inaccurate inverse models for Optical Tomography. The MDM on the other hand produces a reasonable approximation, results from this should provide reasonably quantitative results, particularly as the clear layer reduces in thickness.

7.7 Summary

In this chapter we have shown results from the work we have presented in this thesis. We have shown the RDM to be a valid model for light transport in diffusing domains with non-scattering regions. We have illustrated that its various components are functional and shown our reasons for choosing particular components to be in our final model. Further we have shown the Simplified Meshless Method we developed to be sufficient to our problem and illustrated good results from it. Finally we have compare

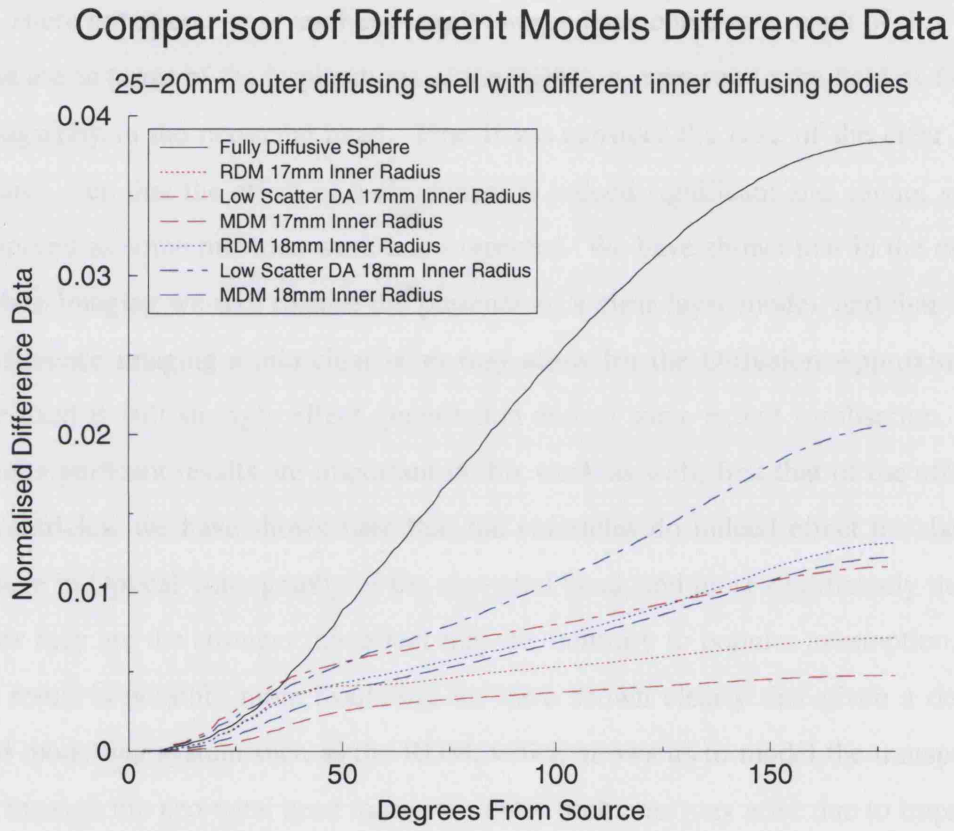


Figure 7.68: A graph showing the comparison of the effects of changing the medium between the RDM, MDM and low-scatter DA.

the diffusive-RDM we presented here with other work in this area and found that our model is comparable to all of them. We showed that simplifications like the MDM may be useful in some circumstances, but that sometimes we may need more sophisticated models like the RDM. In the next chapter we will go on to discuss these results more fully and outline their implications. We will also discuss avenues for further work that these results have brought up and finally summarise the achievements we have made in this thesis.

There are of course several key results which have come as a result of this work. These are in terms of the implications of the RDM once proven to the field of Optical Tomography in the neo-natal head. First if we consider the case of the clear layer, we have seen that the effect of a clear-layer is indeed significant and cannot simply be ignored as some previous work has suggested. We have shown that in the case of absolute imaging we will require the presence of a clear layer model, and that whilst in difference imaging a thin clear layer may allow for the Diffusion Approximation to be used it will strongly effect quantitation and to some extent localisation. Two further significant results are important in this work as well, first that of the effect of the ventricles, we have shown here that the ventricles do indeed effect the data we measure in Optical Tomography of the neo-natal head, and most significantly that the deeper they are the stronger this effect may be, contrary to popular assumption. The final result is possibly more troubling, we have shown clearly that given a domain based modelling system such as the RDM, which allows us to model the transport of light through the neo-natal head more efficiently problems may arise due to imprecise knowledge of the geometry. In fact at this time we suggest that the level of precision required for modelling the clear layers may in fact imply that we require the use of difference data in order to provide reasonable images. These results will be discussed further in the next chapter (chapter 8).

Chapter 8

Conclusions

In this thesis we have introduced a novel 3D model for handling light transport in diffusing domains containing non-scattering spaces. We have shown a hybridisation of methods from optical and neutron transport theory with methods taken from computer graphics. We have illustrated that we have a valid working model by comparison with numerical and physical models and used this to show how the clear regions in the head will effect neo-natal imaging.

Here we will outline the major conclusions of this work, summarising in four major sections. First a look at the model itself, second an examination of its components and various novel aspects developed in this work, third a comparison with other models from the literature and finally a look at the implications of the effects, of such clear regions in the head, have for Optical Tomography of the neo-natal head. After this we will outline some of the major avenues for further work which are brought up by or in consideration with this research. Finally we shall give a brief summary of the chapter indicating how our achievements match our goals.

8.1 The RDM - A Valid Model

In this section we will discuss how we have validated the RDM as a model for light transport in diffusive domains containing non-scattering regions. First we have compared it to a full transport model for a numeric validation, then we have gone on to show comparisons with fields from a selection of physical phantoms. Finally we have gone on to illustrate that the model is well behaved under geometric variation.

8.1.1 A numerically valid model

In the first instance we validated the RDM against an independent full Transport Monte-Carlo model (supplied by Eiji Okada of Keio University). Before doing this we took a visual validation step to ensure the model was working as we would expect, in section 7.1.1 we did this by looking at the outside of a set of spheres with a concentric clear layer of increasing thickness. These results (which we first showed in [71]) clearly show that as the clear layer thickens the light travels round the sphere further and faster precisely as we expect.

We continued to a full numerical validation against the Monte-Carlo Model, section 7.1.2. This showed the RDM to be in agreement with Monte-Carlo simulation, it also showed the failure of the Diffusion Approximation in this case if we set the scatter close to zero ($\mu_s = 10^{-7} mm^{-1}$). We can clearly see that numerically the RDM can match the transport model, whereas in the absence of a clear layer, or even an attempt to simulate one with realistic parameters the DA cannot match the boundary data.

Finally in an effort to understand the problem the Diffusion Approximation had, we compared the visual internal fields of the RDM and the DA using a realistic scatter value. We clearly see in section 7.1.3 the DA providing too much light piping around the void.

From this we can conclude that the RDM is more numerically accurate and physically sensible in the non-scattering inclusions than a Diffusion Approximation. This is not unexpected as we know the Diffusion Approximation should fail in this instance, more significantly though we have seen that the RDM does provide a valid approximation to the RTE for light transport in scattering domains containing non-scattering regions.

8.1.2 A physically valid model

Having shown that we have a model which is numerically valid compared to the RTE for Light Transport, we have gone on to show the RDM to be valid in a physical sense in this case. In order to do this experiments we have presented results of physical experiments conducted with the MONSTIR system.

We have shown comparisons of the Mean-Time of flight of photons in a set of phantoms to the data obtained via numerical simulation with the RDM. We started

by illustrating results using a simple phantom made of resin, glass and intralipid and gradually progressed to a full resin phantom. The progression has been useful in highlighting a piece of future work, though originally it was purely a case of building up from a simple phantom to construct to more correct phantoms required to validate the model.

In section 7.1.4, we started with the simplest phantom, this was a resin beaker of known optical properties with a glass round-bottomed flask inserted to form the void. This set-up was filled by intralipid of matching absorption and scatter to the resin. This provided results which failed to match the numerical data. This however was caused by a mismatch in the refractive index. In a second experiment the flask was filled with resin, here the data came much closer to matching the physical data but was still not a true match. We noted at this point that the refractive index mismatch between resin and glass is not as great as that of intralipid to glass, but resin is higher,not lower, than glass. In fact the numerical results were bounded on either side by the physical data. This is significant as it implies we have correctly identified the source of the error, but also that such a mismatch behaves in a consistent way.

In section 7.4.2 we have presented the result of our final physical experiment. Here we have clearly shown a good match between the data from a physical experiment and the RDM. In this experiment we have removed the index mismatch by using a full resin phantom. We may now state that the RDM is not only a valid model for approximating the RTE in diffusing domains with non-scattering regions, but that it is a physically realistic model in these domains.

8.1.3 A geometrically stable model

One question that remains with the RDM that of geometric stability. We know that we wish to have a smoothly varying solution as the geometry of the domain varies.We have already seen that visually this appears to be the case as discussed in the initial visual validation (see the results in section 7.1.1). In section 7.4.1 we illustrated data from a set of concentric layered ellipses, with a fixed width clear layer. Here we gradually extended a sphere in one direction to form a set of gradually changing ellipses, the boundary data around the geometrically constant part of the model was illustrated in section 7.4.1. Here we see the data changing in a smooth fashion, which tells us we

have a geometrically stable model.

8.2 The RDM - Component Comparisons

In the course of this work we have constructed various models for solving the problem, mostly in view of efficiency and accuracy. In this section we discuss these results both in terms of how they effect the RDM, but also how some compare to other results from the literature. To begin with we will discuss the results of the Form Factor models in terms of accuracy against analytic models, their behaviour within the full RDM model and how these results compare with some previous results in the literature. We will move on to discuss the visibility model results, in comparison to the analytic model, how they perform in the RDM and how understanding the errors has led us to a clear understanding of the restrictions these models place on the domains in which they are used. Finally we will discuss the results from the models we have used and developed to reduce the geometric and computational modelling costs, localised p-refinement and simplified meshless model in terms of their effectiveness and how they help us to understand the problem at hand.

8.2.1 Form Factor Models

The Form Factor Model Comparison provides us with some interesting results. It has previously been stated [99] that a point collocation approach is superior to a Galerkin estimate for the purposes of generating Form Factors, however the base-line used here was another point collocation model. In this work we have presented results for the two models against analytic data, providing a more conclusive result. Interest arises though when we insert the two models into our overall RDM simulation where we acquire a result which contradicts our initial results. The results we discuss here are those presented in section 7.2.1.

First we will consider the point-wise data where we compared the point-wise data from our models to an analytic case. Here we see, as we would expect, that the models are very accurate at the point of calculation, but that as we diverge from this point the Galerkin model appears more accurate. This initially suggests a discrepancy from the results of [99], but when we move on to consider the patch-wise comparison to the analytic model from [84], we see the point collocation method proving better than the

Galerkin. This does seem somewhat surprising as we might expect the more detailed representation of the Galerkin model to provide a better result.

The final result in the Form Factor comparison leaves us with another turn in the choice of model for accuracy. It is clear when we compare the results of using both the point collocation and Galerkin models in the RDM to the Monte-Carlo simulation data that the Galerkin model offers a better result. This seems to contradict previous results for the patch comparison and those suggested in [99]. We would suggest this is caused by our use of a Galerkin approximation for the diffusive domain, as the errors from the Galerkin Radiosity model are essentially in the same space we do not increase our error, but as the point collocation approach creates errors in a different space we have a conflict and our solution is not as accurate.

8.2.2 Visibility Models

Examining the data from the visibility models is somewhat more complicated. At first sight both visibility models provide very good results with a small margin of error, focused around the visibility horizons. The results we are discussing here are presented in section 7.2.2.

Examining the initial error estimates

In the OpenGL case this presents as a discretisation issue in terms of resolution of our view panes and also in terms of mesh resolution, but the primary problem is the resolution of the view panes. In the case of the voxel algorithm, the so called errors in comparison to the analytic are brought about by the mesh resolution. If we examine the source of the errors, we see they arise purely from the geometry of the model and are actually correct values for the mesh as opposed to the represented surface.

Comparison within the RDM to the analytic model

Having examined the initial error estimates we see that both models appear to be adequate for our test-case domain (concentric spheres). In order to compare them properly we examined them in comparison to the analytic model within the RDM, see section 7.2.2. We can see from these results that the three models produce well matched results. An encouraging feature is that the variations are predictable, the OpenGL model reduces the transport around the clear layer meaning a darker far-side from the source, whereas the voxel model promotes the transport of light around the clear layer, exactly

as we expect from their respective difference areas from the analytic model.

This tells us we have two functional visibility models, which are restricted to different domains. The voxel visibility model which is applicable in any given polygonal domain for solving the scene visibility (although this is similar to previously developed models, see for example [2]). The second is a novel approach, the OpenGL algorithm, this model relies however on the domain having smoothly varying shape, and is not functional in domains with sharp corners.

8.2.3 Visibility Representations

The final component of the Radiosity model is the concept of visibility representation. We aim to examine the advantages of using the shadow masking scheme in the Radiosity component of the RDM. In section 7.2.2 we answer this by comparing the results of the different shadow masking schemes on the different Form Factor models with respect to a Monte-Carlo simulation. We can clearly see under the point collocation approach to the Form Factors that the use of shadow masking improves the results, however examining the results from the Galerkin approach the results look very similar - it is unclear if there is improvement here due to the small nature of the change.

8.2.4 Localised P-refinement

Localised P-refinement was introduced with the intent of finding a better way of representing the domain, by using higher order elements only where necessary. The results we presented in section 7.3.1 lead us to a few conclusions. First we see that in a 2D Diffusion problem we can use the changing order basis functions to reduce our number of unknowns in trivial geometries. The second test was in 3D with diffusive domains, and again we see the changing order assisting us in simplifying the model. When we move to the RDM though we see that we actually gained very little saving in terms of number of unknowns by introducing the mixed order elements, we are in fact close to the limit of the number of nodes require to obtain correct simulations of the domain with the given meshes. Whilst the mixed order FEM approach has not proved fruitful in reducing the costs of the RDM it it is useful in the DA models.

8.2.5 Simplified Meshless Method

The simplified meshless method (SMM) has proven to be the more useful of our models for reducing the computational costs. Here we have developed a model, that has proven to produce good results. The results in section 7.3.2 show that the SMM can produce results which match the Monte-Carlo simulation and other RDM models well. There is scope for further improvement here by introducing partial integrals of the basis functions for the DA component of the domain on boundary voxels. This would correct the models' geometric assumption by matching the domain and the boundary exactly. At this time this has been left for further work as in our case the current geometric assumption has proved to be sufficient for our purposes. We have developed a novel mathematical approximation to geometric modelling of our domain for the FEM which is sufficient to our purposes.

8.2.6 A choice of components for further analysis

For our continued analysis of the problem then we have opted for the Galerkin Model with shadow masking. This appears to give us the best result, and is also slightly faster to compute than the point collocation approach. In terms of a visibility model we have used analytic models where possible (cylinders/spheres) for speed and accuracy, but in other geometries we have opted for the voxel visibility model as this is the most accurate of our other approaches (which are comparable in terms of speed). In some cases we have also used the SMM approach to modelling the domain. This is largely where generation of suitable tetrahedral meshes is non-trivial or where a suitable quality mesh takes us outside the computational range of available hardware.

8.3 The RDM - Comparison to alternatives

In this section we will compare the RDM to other models available in the literature. The first of these is a more sophisticated model the P_1 RDM, giving an idea if our diffusive RDM, as it stands, is sufficient (by comparison) to the problem at hand. The other models are simpler models (the Modified Diffusion Model and a best fit Low Scatter Diffusion Approximation) which will give us some idea of the necessity for such a complex model. These issues will further be clarified in our particular case of interest, Optical Tomography, in section 8.4 below.

8.3.1 P_1 RDM

In this work we have developed a diffusive boundary model for the RDM. Our starting point is a Diffusion Approximation using a standard FEM framework, it does not therefore contain information about the normal derivative of the radiance in the model. This leads us to a model which is not strictly as accurate as one where we can model the current by using a different approach. In [46, 73, 90], work done in parallel with this research a fuller P_1 boundary condition was proposed.

In section 7.5.1 we compared the 3D P_1-RDM given in [75] and implemented in [90], to the Diffusion-RDM. It is apparent that both models provide a good fit to the amplitude data, whilst not so good a fit to phase data. We note that despite the more correct nature of the P_1-RDM we get a better fit from the Diffusion-RDM. When we consider that the basic model is the Diffusion Approximation to the P_1 system of equations, and the fact that a diffusion based model for the void fits the Monte-Carlo data better, this seems to be the likely explanation.

8.3.2 Modified Diffusion Model

We have also compared the Radiosity Diffusion Model with the Modified Diffusion Model suggested in [13]. We have illustrated a good agreement in the far field, with the RDM and Monte Carlo models as was presented by the authors (see section 7.5.2). However when looking at the near field we see discrepancies at clear layers of $3mm$ thickness, these do gradually reduce as the layer thins to $1mm$ as we would expect from the models assumption that the thickness of the layer is small in comparison to the transport length of the medium.

One significant feature of the MDM is the shape of the curve. The difference between it and the RDM/Monte-Carlo data is the absence of an inflection in the curve. As opposed to a two step-decay shown in the other models (most clearly evident in the thicker $3mm$ clear layer) the MDM shows a monotonic decay. This feature occurs approximately at the angle of the visibility horizon in the boundary data, we can infer from this that the localising effect of the MDM simply removes this effect. We note that in [4] the effect of a low direction sampling in the 2D discrete ordinates code was to inaccurately reflect this inflection in the data. We would suggest therefore that this may prove significant in models attempting to reconstruct clear layers which do not handle

these inflections as they are the simplest apparent geometric information available from the data.

8.3.3 Low-Scatter Diffusion Approximation

We have already seen that trying to correctly model a non-scattering layer by reducing scattering to effectively zero in a Diffusion Approximation causes the expected failure of the model. One point of interest was to see if we could model the Clear Layer by using a best fit scattering value in the clear layer. In section 7.5.2 we presented data for a best-fit low scattering model. Here we showed that in the given meshes choosing a set scatter value did allow the DA to model the RDM type data. We note here that the same absence of the inflection in the data occurs as with the Modified Diffusion Model.

8.3.4 The RDM - a necessary and sufficient model

At this stage we can comment on the necessity for and sufficiency of the model. First we have an indication that the Diffusion-RDM appears sufficient in comparison to its P_1 cousin, it models light transport at least as well if not better at. The second question is whether it is necessary, at this time it would appear that the two simpler models may in fact also be sufficient and are simpler to compute than the RDM. However as we will see later in section 8.4, this is not the complete story on the necessity issue as other important factors will need to be considered as well.

8.4 The RDM - Application to Optical Tomography

In this section we will examine the data which is more pertinent to our application area for the RDM Optical Tomography. Here we will try to examine the RDM and its components as well as models for non-scattering spaces. We will answer more clearly the question of necessity for a model of more computational cost like the RDM and go on to discuss the implications of this model for Optical Tomography as a tool for neo-natal head imaging systems.

8.4.1 A look at the data and the inverse problem

In section 7.6.1 and [71] we have shown the initial results for the inverse problem indicating that such problems can be solved. We note that in this example an inverse crime was committed as the same mesh was used. In section 7.6.1 and [72] we have gone on to show that the proportional change in data is significantly different in the presence

of a clear layer than in its absence, we suggest therefore that whilst results such as those presented in [66] may be possible they cannot ever achieve accurate quantitation unless a void-like model is introduced. Further if we examine the qualitative differences shown in section 7.6.1 and consider more carefully the quantitative differences, we must conclude that our localisation will also not be as good.

This initial result shows us that whilst we can reconstruct for anomalies beneath a clear layer we must use a reasonable model in order to obtain reliable images. This tells us that a diffusive model is insufficient for the inverse problem. In section 8.4.3 below we will consider other void-like models to determine whether the RDM is a necessary model in this case.

8.4.2 Different visibility models and geometric variation

We know we have a reasonably stable model for varying geometries if we maintain the same model throughout the system. One question about this is how anomalies are effected under minor geometric variance. This is important if we wish to establish a models ability to detect objects beneath an unknown clear layer. The question here is twofold, first if we are trying to include the void in a reconstruction and second if we are trying to model the geometry by using a different modality, say MRI data.

The easiest way to consider this is to compare data from two slightly different meshes. We require however that our anomaly is identical to ensure no numerical differences here affect our result. To do this we considered the data obtained from using different visibility models, see section 7.6.3. If we consider the effect of this we can see that it is equivalent to inserting a sub millimetre uncertainty in the visibility horizon. On examination of the proportional change in each data set we get very good agreement as we would expect and plotted together the homogeneous boundary data was very well matched. However we do encounter a difficulty. If we compare the data between visibility models we do not see the same proportional change, in fact the data from the anomaly is swamped. Examining the six blob-in blob-out curves for the three different visibility models we see three pairs of lines. This is not what we would hope to see, we would like to see two pairs of three lines, implying the blob in and blob out data sets were separated rather than the data sets grouping by visibility model. It tells us that whilst we have a stable model, the changes occurring due to geometric variation

are strong in comparison to changes in the background properties.

If we also look at the data produced by different widths of gap and optical sizes of anomaly we can see that whilst the geometric noise does not appear to vary much in magnitude as we change the gap width, the data magnitude, as we already know, increases as the gap width reduces. This implies that whilst we can expect to generate significant noise from absolute images, it should be possible for a given geometry to determine the optical size of object we can expect to see beneath a clear layer. Thus if we are looking for smaller objects than this we know we will need to examine difference data rather than absolute data in order to recover the objects.

8.4.3 Comparing data from other non-scattering models

One question we have asked is whether we need models such as the RDM for the purposes of Optical Tomography. Clearly if we wish to examine absolute data where we have no reference, just examining the data from the validation procedures tells us we must model the clear region in some manner. What is not clear however is whether we will need such models when it comes to examining difference imaging.

In the case of difference imaging we have the situation where we have a background image and an anomaly image, we will assume the clear region is part of the background in the simplest case. Here then some results [66] have been shown where such void models appear to be unnecessary, however preliminary 2D results seem to disagree with this [25]. In this work we have taken a two stage investigation, first we have looked at simple difference data as void width varies, and then we have looked at using simpler models to handle the clear layer. In section 8.4.4 we will go on to look at the more complex issue of introducing void inclusions in the medium (equivalent to ventricles) to see how they effect the data.

The effect of the clear layer on difference data - varying the gap width

There are two ways this data has been presented, qualitatively and quantitatively (section 7.6.1). First we shall examine the qualitative approach, here we clearly see that whilst the data appears to be similar between the different thicknesses of gap layer, as the gap broadens the image becomes more blurred. This would qualify well with results in 2D [25] which suggested that lacking a correct gap model effected the ability to localise an anomaly with difference data. The quantitative results illustrate the other

feature of not using a correct gap model, the change in the data is of a lower relative value than if we have no gap model present, this suggests that in the absence of a void model we will obtain poor qualitative images, also given this we will be able to detect smaller anomalies if we use a correct model.

The effect of the clear layer on difference data - different models

We have already seen that there is a need for a void model in the presence of a clear layer. The obvious question is will a simpler model than the RDM be sufficient to the problem. In section 7.5.2 we compared the difference data from two other simpler void models, the MDM and a best-fit low scatter Diffusion Approximation. The results here suggest that a more complex model may indeed be necessary, we see that in the low scatter DA when we use the best fit scattering value our data does not match in any way the data from a more correct model. In the case of the MDM where we have a model which does approximate the physics, we do get a closer match to the data, we still see a slight drop in data levels produced by a blob, however which reduces as the gap thins.

8.4.4 Including non-scattering inclusions as well as layers

One final aspect we must consider is the effect that clear regions have on the data. In section 7.6.2 we clearly see a need to model the ventricles. Whilst the models geometry is primitive it does illustrate the effects of the ventricles dramatically.

The first thing we notice in the data is that there are now two peaks instead of one, we note that the unevenness in the graph is due to the numerical error produced by the mesh being asymmetric. We also note that as the object passes deeper into the medium the effect of the ventricles on the data is less. What however is notable in these results is the effect on the data as the ventricles deepen, the effect on the data actually becomes more pronounced. This suggests that a full model for the CSF will be required if we are to produce sensible results from OT data, even difference data.

8.4.5 Implications for Optical Tomography

In this section we have discussed various results we have presented and their individual effects on our requirements to model the transport of light in the neo-natal head for the purposes of OT. First we have shown that for the optical case accurate geometric models will be necessary if we are to reconstruct data in the absolute sense. It may

be possible to reconstruct these clear regions, but there exists no uniqueness proof for this problem at this time (in combination with varying parameters). Therefore we have also looked at the effects of clear regions on the OT problem if we are only to consider difference imaging. In this work we have shown that a simple DA is not sufficient and that a fuller model is necessary, we suggest that in such cases we will have more freedom with the geometry provided we use the same model in forward and inverse solutions. The question of whether the MDM is sufficient is unclear, but at this time it is not available in unrestricted geometries and therefore lends us to suggest at this time the RDM is the better model.

8.5 Potential Developments for the RDM

In this work we have illustrated that for our chosen domain type, the RDM is a valid and useful model. In this section we will consider various questions which have been raised during this work as to possible future enhancements of the model.

We will consider first the Refractive Index mismatch problem, an issue raised by the results of attempting to validate the model with physical phantoms. We will then consider what happens if we were to introduce a low-scattering region, one where neither the DA applies nor the RDM.

8.5.1 Refractive Index

We have illustrated clearly during the process of validating this model with three experimental set-ups that we will need to handle changes in the refractive index if we are to model domains of this type. It is clear from the results that such changes do appear to have a very structured effect on the model and so should be modelable in some relatively simple manner. Current suggestions include adding a ray-tracing element to the void model alongside the Fresnel coefficients. Such a model would either involve some kind of recursive model, or a model which included some 3-way modelling of the Form Factors, as we must know not just one direction but two, as we no longer have an even BRDF. In fact recently an initial study has been published showing how this may be achieved based on an RDM type model, see [56]

8.5.2 Low-Scattering Model

Some authors [33] have raised a question as to whether or not the CSF is truly non-scattering. It is unclear at this time whether the membranous components (trabeculae) of the sub-arachnoid space will make the clear layer locally partly scattering. The evidence previously presented on this [33] compares a selection of numerical models to experimental data acquired from a topographic set-up on an adult head, these comparisons however are flawed in the absence of a correctly matching geometry, the absence of a refractive index mismatch and finally the assumption that all other tissue values are known and Diffusive. At this time we would suggest that whilst this remains unclear modelling should try to deliver both possibilities.

We note there are a selection of ways to proceed with this approach. First we could use a different model in the low-scattering regions, integrating out the scattered light and treating it as a source term within the clear regions. Other suggestions would include using fuller transport models in this region. We have examined the possibility of extending the RDM by using work of [78] where a full transport model (called the zonal method) was inserted inside a radiosity domain to handle scattering spaces. We note that this model produces singularities in the system of equations, and at this time it is unclear how to handle these singularities. This path was examined as it seemed a logical extension to the current regime. It may be possible to further examine this problem to handle the singularities and complete this model. We note however that this approach does lead to an even more dense problem than the original RDM, typical to fuller transport solutions and would require significantly more processing power for realistically sized domains than the current models.

8.6 Summary

In this work we have developed a new approach to modelling Light Transport in 3D. We have shown that for diffusing domains containing non-scattering spaces we can, rather than using a full transport model as previous work has done, use simplifying approximations in each part of domain. Taking work in classical transport literature, and work from computer graphics we have developed a working direct 3D model for light transport in such domains.

We have illustrated that this model does provide numerically and physically valid

results for such domains, allowing us to use it to investigate the modelling of such problems in a case study of Optical Tomography. We have shown that such models are necessary to modelling the domain as simple diffusion models are incapable of handling such domains.

We have illustrated that the model is stable to gradual physical change in the domain, although we have shown that to study changes in the medium does require accurate geometric modelling. The implications here are that whilst the model is sufficient to study the inverse problem this is only true in the case where we have a background and an anomaly image.

We have shown it is favourable in comparison to other models for the domain in correctly measuring the light fields. This is important as it allows us to use this model as a baseline to assess the usefulness of some simpler approximations, such as the DA, which was shown to be lacking in domains with non-scattering spaces and the MDM, a model which if correctly extended may be useful for modelling but currently is limited to certain geometries. We have also illustrated that the Diffusion RDM gives better results than the P_1 RDM in modelling the domain, we assert that this is due to matching the approximation used in the rest of the domain.

On the way to handling such domains we have tried various approaches to handling the modelling. We have developed a novel visibility algorithm using OpenGL, which is limited strictly to smoothly varying domains. In the field of Form Factor modelling we have confirmed earlier results suggesting that in a patch-wise sense that point-collocation approaches appear to be superior to Galerkin models, but that in our case as we use a Galerkin estimate for the rest of our domain we are better using a matched (Galerkin) error estimate for the Radiosity boundary condition as well.

We have explored various avenues to improve the problem of computational costs incurred when we have complex domains which cause problems with large mesh sizes. We have shown that localised p-refinement can be useful in certain domains, although it was unsuitable to the problem of the RDM. More importantly we have developed a novel simplified meshless model, whose geometric assumptions are sufficient to our domain.

In terms of Optical Tomography in the presence of clear regions, i.e. on the neonatal head, we have shown some initial results. We clearly see that modelling the clear

regions is essential, not just to absolute imaging but also to difference imaging, contrary to some of the current literature, if we are to obtain quantifiable results. We have illustrated that in the presence of such clear regions we must have good resolution of the geometry if we are to obtain good images of objects beneath the clear layer from absolute imaging, but that such structures will not be as severe if we use difference imaging. We also mention that in the real case the RDM would have to be extended to take proper account of the Refractive Index mismatch (as shown in our physical validation), further we suggest that better models will be required to handle the other non-diffusing tissue types in the head, such as the anisotropic regions of the white matter. Whilst work is being done in these areas we would suggest it may be some time before a reliable model of light transport for the full head problem will be developed. We suggest the RDM however does show the way by merging appropriate approximations for each sub-region, rather than resorting to more expensive full models throughout the domain.

To conclude we have produced a valid model for light transport in Diffusive domains containing non-scattering regions. We have shown this model to be numerically equivalent to more expensive full transport solutions and also to match data from a physical phantom. We have shown how these clear regions will effect the data in the case of Optical Tomography and showed that models such as the RDM will be necessary if we are to make images of the neo-natal head with Optical Tomography.

Bibliography

[1] R. T. Ackroyd. *Finite Element Methods for Particle Transport.* Research Studies Press Ltd., 1997.

[2] J. Amanatides and A. Woo. A fast voxel traversal algorithm for ray tracing. In *Proceedings of Eurographics '87*, pages 1–10, 1987.

[3] S. R. Arridge. Optical tomography in medical imaging. *Inverse Problems*, 15:p41–R93, 1999.

[4] S. R. Arridge, H. Dehghani, M. Schweiger, and E. Okada. The finite element model for the propagation of light in scattering media : A direct method for domains with non-scattering regions. *Med. Phys.*, 27:252–264, 2000.

[5] S. R. Arridge and W. R. B. Lionheart. Nonuniqueness in diffusion-based optical tomography. *Opt. Lett.*, (23):882–884, 1998.

[6] S. R. Arridge, M. Schweiger, M. Hiroka, and D. Delpy. A finite element approach for modelling photon transport in tissue. *Med. Phys.*, 20(2):p299–309, 1993.

[7] J. Arvo. Transfer equations in global illumination. *SIGGRAPH '93 Course Notes*, 42, 1993.

[8] E. Aydin, C. Oliveira, and A. Goddard. A comparison between transport and diffusion calculations using a finite element-spherical harmonics radiation transport method. *Med. Phys.*, 29(9):2013–2023, 2002.

[9] I. Babuska and A. Aziz. On the angle condition in the finite element method. *SIAM Journal on Numerical Analysis*, 13, 1976.

[10] I. Babuska, U. Banerjee, and J.E. Osborn. Meshless and generalized finite element methods: A survey of some major results. Technical report, Institute for Computational and Engineering Sciences, January 2002. TICAM Report 02-03.

[11] I. Babuska and Suri. The p and h-p versions of the finite element method, basic principles and properties. *SIAM Review*, 36(4):578–632, 1992.

[12] G. Bal. Particle transport through scattering regions with clear layers and inclusions. *J. Comp. Phys.*, 180(2):659–685, 2002.

[13] G. Bal and K.Rui. Generalized diffusion model in optical tomography with clear layers. *Journal of the Optical Society of America A*, 20(12):2355–2364, December 2003.

[14] B. Balendran. A direct smoothing method for surface meshes. In *8th International Meshing Roundtable*, pages 189–193, 1999.

[15] A. Becker. *The Boundary Element Method in Engineering. A complete course.* McGraw-Hill Book Company, 1992.

[16] M. Born and E. Wolf. *Principles of Optics.* Cambridge University Press, 7th(expanded) edition, 1999.

[17] C. Brechbüler, G. Gerig, and O. Kubler. Parameterisation of closed surfaces for 3d shape description. *Computer Vision and Image Understanding*, 61(2):154–170, 1995.

[18] J. E. Bresenham. Algorithm for computer control of a digital plotter. *IBM Systems Jounral*, 4(1):25–30, 1965.

[19] K. Case and P. Zwiefel. *Linear Transport Theory.* Addison Wesley, 1967.

[20] S. Chandreskar. *Radiative Transfer.* Dover Publications inc., New York, second edition, 1960.

[21] P. Chew. Guaranteed-quality mesh generation. In *Proceedings of the 9th annual symposium on Computational Geometry*, pages 274–280, 1993.

[22] M. Cohen, S. E. Chen, J. R. Wallace, and D. P. Greenburg. A progressive refinement approach to fast radiosity image generation. *Computer Graphics*, 22:75–84, 1988.

[23] M. Cohen and D. Greenburg. The hemi-cube: A radiosity solution for complex environments. *ACM Computer Graphics*, 19(3):31–40, 1985.

[24] M. Cohen and J. Wallace. *Radiosity and Realistic Image Synthesis*. Academic Press Professional, Boston, MA, 1993.

[25] H. Dehghani, S. R. Arridge, M. Schweiger, and D. T. Delpy. Optical tomography in the presence of void regions. *J. Opt. Soc. Am. A*, 17:1659–1670, 2000.

[26] O. Dorn. A transport-backtransport method for optical tomography. *Inverse Problems*, 14:1107–1130, 1998.

[27] O. Dorn. Scattering and absorption transport sensitivity functions for optical tomography. *Opt. Express*, 7,13:492–506, 2000.

[28] G. Drettakis and F. Sillion. Accurate visibility and meshing calculations for hierarchical radiosity. In *Proceedings of Seventh Eurographics Workshop on Rendering*, June 1996.

[29] I. Driver, J. W. Feather, P. R. King, and J. B. Dawson. The optical properties of aqueous suspensions of intralipid, a fat emulsion. *Phys. Med. Biol.*, 34(12):1927–1930, 1989.

[30] J. Duderstadt and W. R. Martin. *Transport Theory*. Wiley-Interscience, 1979.

[31] F. Durand, G. Dretakis, and C. Puech. Fast and accurate hierarchical radiosity using global visibility. *ACM Transactions on Graphics*, 18(2):128–170, 1999.

[32] F. Durand, G.Dretakis, and C. Puech. The visibility skeleton: A powerful and efficient multi-purpose global visibility tool. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Technology*, pages 89–100, 1997.

[33] E.Okada and D. T. Delpy. Effect of scattering of arachnoid trabeculae on light propagation in the adult brain. OSA Technical Digest,Biomedical Topical Meetings (OSA, Washington DC).

[34] M. Feixas, E. Acebo, and M. Sbert. Entropy of scene visibility. *WSCG*, 1999.

[35] M. Firbank, M. Oda, and D.Delpy. An improved design for a stable and reproducible phantom material for use in near-infrared spectroscopy and imaging. *Phys. Med. Biol.*, 40:955–961, 1995.

[36] J. K. Fletcher. The solution of the multigroup neutron transport equation using spherical harmonics. *Nuclear Science and Engineering*, 84:33–46, 1983.

[37] J. K. Fletcher. A solution of the multigroup neutron transport equation using spherical harmonics. *Transport Theory and Statistical Physics*, 15:157–179, 1986.

[38] J. Foley, A. van Dam, S. Feiner, J.Hughes, and R. Phillips. *Introduction to Computer Graphics*. Addison Wesley Publishing Company, 1986.

[39] I. Fried. Condition of finite element matrices generated from non-uniform meshes. *AIAA Journal*, (10):219–221, 1972.

[40] H. Fuchs, Z. M. Kedem, and B. F. Naylor. On visible surface generation by a priori tree structures. *Computer Graphics*, 14(3):124–133, 1980.

[41] A. Gibson, R. Md. Yusof, H. Dehghani, J. Riley, N. Everdell, R. Robins, J. C. Hebden, M. Schwieger, S. R. Arridge, and D. T. Delpy. Optical tomography of a realistic neonatal head phantom. *Applied Optics*, 42:3109–3116, 2003.

[42] A. P. Gibson, J. C. Hebden, J. Riley, N. Everdell, M. Schweiger, S. R. Arridge, and D. T. Delpy. Linear and non-linear reconstruction for optical tomography of phantoms with non-scattering regions. Submitted to Applied Optics.

[43] A. H. Hielscher, R. E. Alcouffe, and R. L. Barbour. Comparison of finite-difference transport and diffusion calculations for photon migration in homogeneous and heterogeneous tissue. *Phys. Med. Biol.*, 43:1285–1302, 1998.

[44] E. M. C. Hillman. *Time-Resolved Diffuse Optical Tomography of the Breast and Brain.* PhD thesis, University College London, 2002.

[45] K. Höllig. *Finite Element Methods with B-Splines.* SIAM, 2003.

[46] N. Hyvönen. Analysis of optical tomography with non-scattering regions. Master's thesis, Helsinki University of Technology, 2000.

[47] A. Ishimaru. *Wave Progation and Scattering in Random Media:1.* Academic Press, Orlando, Florida, first edition, 1978.

[48] A. Ishimaru. *Wave Progation and Scattering in Random Media:2.* Academic Press, Orlando, Florida, first edition, 1978.

[49] H. Van Jensen. *Realistic Image Synthesis Using Photon Mapping.* AK Peters, 2001.

[50] G. Jones. *Accurate Radiosity Methods for Computer Graphics.* PhD thesis, Oxford University, 1999.

[51] J. T. Kajiya. The rendering equation. *Computer Graphics*, 20(4):143–150, 1986.

[52] A. D. Klose and A. H. Hielscher. Iterative reconstruction scheme for optical tomography based on the equation of radiative transfer. *Med. Phys.*, 26:1698–1707, 1999.

[53] P. Krysl and M. Ortiz. Generation of tetrahedral finite element meshes: Variational Delauney approach. In *Proceedings of the 7th Meshing Roundtable*, pages 273–284, 1998.

[54] K. D. Lathrop. Ray effects in discrete ordinates equations. *Nuclear Science and Engineering*, 32:357–369, 1968.

[55] K. D. Lathrop. Remedies for ray effects. *Nuclear Science and Engineering*, 45:255–268, 1971.

[56] J. H. Lee, S. Kim, and Y. T. Kim. Diffuse-diffuse photon coupling via nonscattering void in the presence of refractive index mismatch on the void boundary. *Med. Phys.*, 31(8):2237–2248, August 1994.

[57] E. Lewis and W. Miller. *Computational Methods in Neutron Transport.* J. Wiley and Sons, 1984.

[58] N. Max. Efficient light propagation for multiple anisotropic volume scattering. In *Proceedings 5th Eurographics Workshop on Rendering*, June 1994.

[59] W. F. Miller(Jr.), E. E. Lewis, and E. C. Roscow. The application of phase-space finite elements to the one dimensional neutron transport equation. *Nuclear Science and Engineering*, 51:148–156, 1973.

[60] W. F. Miller(Jr.), E. E. Lewis, and E. C. Roscow. The application of phase-space finite elements to the two-dimensional neutron transport equation in x-y geometry. *Nuclear Science and Engineering*, 52:12–22, 1973.

[61] J. Munro, M. H. Overmars, and D. Wood. Variations on visibility. In *Proceedings of the 3rd annual Symposium on Computational Geometry*, pages 291–299, 1987.

[62] L. Neumann and A. Neumann. Radiosity and hybrid methods. *ACM Transactions on Computer Graphics*, 14(3):233–265, 1995.

[63] E. Okada, M. Schweiger, S. R. Arridge, M. Firbank, and D. T. Delpy. Experimental validation of monte carlo and finite-element methods of estimation of the optical path length in inhomogeneous tissue. *Applied Optics*, 35:3362–3371, 1996.

[64] C. Olivier-Gooch. http://tetra.mech.ubc.ca/grummp/. Last accesed February 2001.

[65] S. Owen. Meshing research corner. http://www.andrew.cmu.edu/user/sowen/mesh.html, Last accesed March 2004.

[66] Y. Pei, H. L. Graber, and R. L. Barbour. Normalized-constraint algorithm for minimizing inter-parameter crosstalk in dc optical tomography. *Opt. Express*, 9,2:97–109, 2001.

[67] M. Pellegrini. Rendering equation revisited:how to avoid explicit visibility computations. In *Proceedings 10th Annual ACM-SIAM symposium on Discrete Algorithms*, pages 725–733, 1999.

[68] F. Perez, X. Pueyo, and F. Sillion. Global illumination techniques for the simulation of participating media. In J. Dorsey and P. Slusallek, editors, *Rendering Techniques 97*, pages 309–320. Springer Verlag, Wien, 1997. Proceedings of Eigth Eurographics Workshop on Rendering (Saint-Etienne, France, June 1997).

[69] J. Riley. Light propagation in scattering and non-scattering spaces. Master's thesis, University College London, 1999.

[70] J. Riley, S. Arridge, Y. Chrsyanthou, E. Hillman, H. Dehghani, and M. Schweiger. The radiosity diffusion model in 3d. In *Proceedings of the European Conference on Bio-Medical Optics*, volume 4431. OSA, June 2001. Photon Migration, Optical Coherence Tomography, and Microscopy.

[71] J. Riley, H. Dehghani, M. Schweiger, S. R. Arridge, J. Ripoll, and M. Nieto-Vesperinas. 3d optical tomography in the presence of void regions. *Opt. Express*, 7,13:462–467, 2000.

[72] J. Riley, E. M. C. Hillman, J. C. Hebden, and S. R. Arridge. Light transport in scattering domains containing non-scattering spaces. In *International Workshop - "Comptational Problems of Electrical Engineering"*, pages 215–218, Zakopane,Poland, 2002.

[73] J. Ripoll. *Light Diffusion in Turbid Media with Biomedical Application.* PhD thesis, Faculty of the Univesidad Autonoma de Madrid, 2000.

[74] J. Ripoll, S. R. Arridge, H. Dehghani, and M. Nieto-Vesperinas. Boundary conditions for light propagation in diffusive media with nonscattering regions. *J. Opt. Soc. Am. A*, 17:1671–1681, 2000.

[75] J. Ripoll, M. Nieto-Vesperinas, S. R. Arridge, and H.Dehghani. Boundary conditions for light propagation in diffusive media with nonscattering regions. *JOSA*, 17(9):1671–1681, 2000.

[76] H. Rushmeier. Rendering participating media: Problems and solutions from application areas. In *Proceedings of the FiftEurographics Rendering Workshop*, Darmstadt, Germany, June 1994. Springer-Verlag. Photorealistic Rendering Technique.

[77] H. Rushmeier. Rendering participating media: Problems and solutions from application areas. In *Proceedings of the FiftEurographics Rendering Workshop*, Darmstadt, Germany, June 1994. Springer-Verlag. Photorealistic Rendering Technique.

[78] H. Rushmeier and K. Torrance. The zonal method for calculating light intensities in the presence of a participating medium. *Computer Graphics*, 21(4), 1987.

[79] J. Schewchuk. *Delauney Refinement Mesh Generation.* PhD thesis, CMU, 1997.

[80] F. E. W. Schmidt, M. E. Fry, E. M. C. Hillman, J. C. Hebden, and D. T. Delpy. A 32-channel time-resolved instrument for medical optical tomography. *Review of Scientific Instruments*, 71(1):256–265, 2000.

[81] J. Schöberl. Netgen. http://www.sfb013.uni-linz.ac.at/ joachim/netgen/, Last accesed December 2004.

[82] J. Schöberl. Netgen - an advancing front 2d/3d-mesh generator based on abstract rules. *Comput.Visual.Sci.*, 1:41–52, 1997.

[83] P. Schroder and P. Hanrahan. Analytic form factor, Mathematica code and C library, 1993. http://www.multires.caltech.edu/pubs/, Last accesed January 2003.

[84] P. Schroder and P. Hanrahan. A closed expression for the form factor between two polygons. Technical Report CS-404-93, Department of Computer Science, Princeton University, 1993.

[85] M. Schweiger and S. Arridge. http://www.medphys.ucl.ac.uk/research/borg/toast/index.htm. Last accesed January 2004.

[86] M. Schweiger, S. R. Arridge, M. Hiroka, and D. Delpy. The finite element model for the propagation of light in scattering media: Boundary and source conditions. *Med. Phys.*, 22(11):p1779–1792, 1995.

[87] S. J. Sherwin. Hierachical hp finite elements in hybrid domains. *Finite Elements in Analysis and Design*, 27:109–119, 1997.

[88] K. Shimada. *Physically based mesh Generation.* PhD thesis, MIT, 1993.

[89] H. Si. http://tetgen.berlios.de/. Last accesed December 2000.

[90] J. Sikora, J. Riley, S. Arridge, and J.Ripoll. Analysis of light propagation in diffusive media with non-scattering regions using 3D BEM. Technical report, University College London, 2003.

[91] F. Sillion. A unified hierarchical algorithm for global illumination with scattering volumes and object clusters. *IEEE Transactions on Visualisation and Computer Graphics*, 1(3), September 1995.

[92] F. Sillion and G. Dretakis. Feature-based control of visibility error: A multi-resolution clustering algorithm for global illumination. In *Proceedings of the 22nd annual ACM conference on Computer graphics*, pages 145–152, 1995.

[93] F. Sillion and C. Puech. *Radiosity and Global Illumination.* Morgan Kaufmann Publishers inc., 1994.

[94] M. Slater, A. Steed, and Y. Chrysanthou. *Computer Graphics and Virtual Environments: From Realism to Real-Time.* Addison Wesley, 2002.

[95] J. Stam. *Multi-Scale Stochastic Modelling of Complex Natural Phenomena.* PhD thesis, Dept. of Computer Science, University of Toronto, 1995.

[96] J. Stam. Multiple scattering as a diffusion process. In *Proceedings of the 6th Eurographics Workshop on Rendering*, Dublin, Ireland, June 1995.

[97] J. Stam and E. Fiume. Depicting fire and other gaseous phenomena using diffusion processes. In *Proceedings of the 22nd annual ACM conference on Computer Graphics*, 1995.

[98] J. F Thompson, B. K. Soni, and N. P. Weatherill, editors. *Handbook of Mesh Generation.* CRC Press, 1998.

[99] R. Troutman and N. Max. Radiosity algorithms using higher order finite element methods. In *Proceedings of the 20th annual conference on Computer graphics*, pages 209–212, 1993.

[100] S. Vavasis. http://www.cs.cornell.edu/home/vavasis/qmg2.0/qmg2_0_home.html. Last accesed February 2001.

[101] L. Wang, S. Jaques, and L. Zheng. MCML - Monte Carlo modeling of light transport in multi-layered tissues. *Computer Methods and Programs in Biomedicine*, 47:131–146, 1995.

[102] Z. X. Wang and D. R. Guo. *Special Functions*. Singapore: World Scientific, 1989.

[103] T. C. Warburton, S. J. Sherwin, and G. E. Karaniadakis. Basis functions for triangular and quadrilateral high-order elements. *SIAM Journal of Scientific Computing*, 20(5):1671–1695, 1999.

[104] P. L. Williams. Visibility ordering meshed polyhedra. *ACM Transactions on Graphics*, 11(2):103–126, April 1992.

[105] S. Wright. A note on solving the radiative transfer equation in the vicinity of voids and sources. Technical report, University College London, 2003.

[106] A. Zacharpoulos. Three-dimensional shape-based reconstruction in medical imaging. Master's thesis, University College London, 2003. MPhil Transfer Thesis.

[107] H. R. Zatz. Galerkin radiosity: A higher order solution for global illumination. Master's thesis, Cornell University, 1993.

[108] O. C. Zienkiewicz and R. L. Taylor. *The Finite Element Method*, volume 1. McGraw Hill, fourth edition, 1989.

[109] D. G. Zill and M. R. Cullen. *Advanced Engineering Mathematics*. PWS-Kent, 1992.

[110] G. Zumbusch. Simultaneous h-p adaption in multilevel finite elements, 1996.